

Deep Learning for Aerial Scene Understanding in High Resolution Remote Sensing Imagery from the Lab to the Wild

Yuansheng Hua

Vollständiger Abdruck der von der TUM School of Engineering and Design der Technischen Universität München zur Erlangung des akademischen Grades eines Doktors der Ingenieurwissenschaften genehmigten Dissertation.

Vorsitzender:

Prof. Dr. Urs Hugentobler

Prüfende der Dissertation:

1. Prof. Dr.-Ing. habil. Xiao Xiang Zhu
2. Prof. Dr.-Ing. Lichao Mou
3. Prof. Dr. Sébastien Lefèvre

Die Dissertation wurde am 05.01.2022 bei der Technischen Universität München eingereicht und durch die TUM School of Engineering and Design am 14.06.2022 angenommen.

Abstract

The substantial progress of remote sensing technologies makes aerial images available in large numbers, which benefits a variety of applications, such as urban planning and terrain surface analysis. As a fundamental bridge between such applications and high resolution aerial imagery, aerial scene understanding has attracted increasing attention over the past few years, and manifold efforts have been made from three perspectives: scene-level, object-level, and pixel-level understanding of aerial scene images. Specifically, scene-level understanding refers to recognizing scene categories of aerial images, and object-level understanding is to identify all co-occurring objects in each image. Besides, pixel-level aerial scene understanding is achieved by identifying the semantic class of every single pixel and producing a semantic mask for each aerial scene image. These three levels are termed as aerial scene recognition, multi-label object classification, and semantic segmentation of aerial imagery in this dissertation.

Thanks to the revolutionary progress made by deep learning, great achievements have been obtained in aerial scene understanding. However, most of the existing researches are conducted under the laboratory circumstance, where constraints are imposed on experimental prerequisites and data preparation. As a consequence, the deployment of deep learning models in the wild is a severe predicament, as abundant training samples with precise and complete annotations are scarcely available. To understand aerial scenes and take a step beyond the laboratory scenario, this dissertation makes contributions from the following perspectives:

- For a holistic object understanding of aerial scene images, we propose to encode underlying label correlations with a bidirectional LSTM and a relational network and devise two multi-label object classification networks, respectively. In addition, two multi-label aerial image datasets are proposed to facilitate the progress of object-level scene understanding.
- As an attempt to bridge gaps between aerial scene recognition in the lab and wild, we propose to learn prototype representations of aerial scenes from numerous labeled constrained images, and predict unconstrained aerial images by measuring relevances between images and scene prototypes.
- To address data efficiency in multi-scene recognition, we propose a large-scale dataset, MultiScene, for multi-scene recognition in single images, which is featured by unconstrained multi-scene aerial images and the available both crowdsourced and clean labels.
- In order to train aerial scene parsing models with sparse annotations, we propose an annotation-friendly framework, where annotators only need to label several pixels with easy-to-draw scribbles. To exploit these sparse scribbled annotations, feature and spatial relationships among pixels are encoded for semi-supervised learning of semantic segmentation networks.

Zusammenfassung

Der große Fortschritt der Fernerkundungstechnologien macht Luftbilder in großer Zahl verfügbar, was einer Vielzahl von Anwendungen zugutekommt, beispielsweise der Stadtplanung und der Geländeanalyse. Als grundlegende Brücke zwischen solchen Anwendungen und hochauflösenden Luftbildern hat das Verständnis von Luftaufnahmen in den letzten Jahren zunehmende Aufmerksamkeit auf sich gezogen, und es wurden vielfältige Anstrengungen aus drei Perspektiven unternommen: Verständnis auf Szenen ebene, Objektebene und Pixelebene von Luftbildaufnahmen. Insbesondere bezieht sich das Verständnis auf Szenen ebene auf das Erkennen von Szenekategorien von Luftbildern, und das Verständnis auf Objektebene besteht darin, alle gleichzeitig auftretenden Objekte in jedem Bild zu identifizieren. Außerdem wird ein Verständnis von Luftszenen auf Pixelebene erreicht, indem die semantische Klasse jedes einzelnen Pixels identifiziert und eine semantische Maske für jedes Luftszenenbild erzeugt wird. Diese drei Ebenen werden in dieser Dissertation als Luftszeneerkennung, Multi-Label-Objektklassifizierung und semantische Segmentierung von Luftbildern bezeichnet.

Dank der revolutionären Fortschritte, die durch Deep Learning erzielt wurden, wurden zahlreiche Errungenschaften beim Verständnis von Luftbildszenen erzielt. Die meisten der bestehenden Forschungen werden jedoch unter Laborbedingungen durchgeführt, wobei den experimentellen Voraussetzungen und der Datenaufbereitung Beschränkungen auferlegt werden. Infolgedessen ist der Einsatz von Deep-Learning-Modellen in freier Wildbahn eine ernste Notlage, da kaum Trainingsbeispiele mit präzisen und vollständigen Annotationen zur Verfügung stehen. Um Luftaufnahmen zu verstehen und einen Schritt über das Laborszenario hinauszugehen, liefert diese Dissertation Beiträge aus folgenden Perspektiven:

- Für ein ganzheitliches Objektverständnis von Luftszenenbildern schlagen wir vor, zugrundeliegende Label-Korrelationen mit einem bidirektionalen LSTM und einem relationalen Netzwerk zu kodieren und jeweils zwei Multi-Label-Objektklassifizierungsnetzwerke zu konzipieren erleichtern den Fortschritt des Szeneverständnisses auf Objektebene.
- Als Versuch, die Lücken zwischen der Luftbilderkennung im Labor und in der Wildnis zu schließen, schlagen wir vor, Prototypdarstellungen von Luftbildszenen aus zahlreichen gekennzeichneten eingeschränkten Bildern zu lernen und uneingeschränkte Luftbilder vorherzusagen, indem die Relevanz zwischen Bildern und Szeneprototypen gemessen wird.
- Um die Dateneffizienz bei der Mehrszenenerkennung zu verbessern, schlagen wir einen großen Datensatz, MultiScene, für die Mehrszenenerkennung in Einzelbildern vor, der durch uneingeschränkte Mehrszenen-Luftbilder und die verfügbaren sowohl Crowdsourcing als auch Clean Labels gekennzeichnet ist.
- Um Luftbildszenen-Parsing-Modelle mit spärlichen Annotationen zu trainieren, schlagen wir ein annotationsfreundliches Framework vor, bei dem Annotatoren nur

Zusammenfassung

mehrere Pixel mit einfach zu zeichnenden Scribbles beschriftet müssen kodiert für halbüberwachtes Lernen von semantischen Segmentierungsnetzwerken.

Acknowledgements

Eventually, standing at the end of the long journey, I would appreciate many many people, especially those who accompany and support me either physically or mentally. There are countless happy and depressive moments. There are unforgettable successes and failures. There are figures joining in and out. There are impressive and beautiful landscapes along the trip. There are also adventures and challenges hindering the progress. But finally, here is the end of my Ph.D. candidate journey, and I will do my best to thank everyone before the curtain call.

I would like to sincerely appreciate my supervisor (“Doktoreltern”), Prof. Xiao Xiang Zhu, for her granting me the opportunity of pursuing a doctoral degree and always kind and great support during my research period. As a professor and group leader, her professional insights, broadened horizons, and efficient team management set an example to me of how to be not only a qualified researcher but also a responsible person in daily life. On the stage, she gave impressive talks in TED x TUM and keynote reports of various international conferences, drawing a holistic picture of now and future of the community. Behind the scene, she kindly supports the progress of Ph.D. and master students by raising insightful suggestions and critical comments. I still remember her encouragement in my first paper as well as praise for my last paper. Besides, she often provides group members with opportunities of attending international conferences and showcasing their researches to the community. In daily life, she is friendly and releases the pressure of group members by organizing outdoor activities, big meals, and having relaxing communications.

I would like to sincerely thank my “mentor”, Dr. Lichao Mou, for his mentoring in research and life. The beginning of our knowing each other is that I coincidentally picked the master thesis topic offered by him during my postgraduate study. Since then, we’ve been teacher-student as well as friends. He teaches me the difference between engineering and scientific thinking logic, which does help me with avoiding detours on the research pathway. Recalling the first time I got feedback from him, I was impressed that not only logic mistakes but also inaccurate use of grammar and punctuation are revised, and numerous popping-up comments in PDF even slowed down my laptop. Besides, he is patient and always available for questions and confusion. I even remember that we discussed online paper revision over the whole night. Except for mentoring research issues, he also pulled me back from the edge of the cliff by figuring out my frequent carefulness and demotivation with both encouraging and critical words.

I appreciate Prof. Sébastien Lefèvre for his kindness of being one of my thesis committee members. I am also inspired by his researches in semantic segmentation, which provide me with new perspectives of tackling challenges of segmenting high resolution remote sensing imagery. I also would like to thank Prof. Devis Tuia for his temporary supervision during my exchange to Wageningen University. He showed me his rich knowledge and kindness, making me feel at home during the period of my exchange. I did enjoy the collaboration with him and his group members, Diego Marcos and Sylvain Lobry, and learned advanced research and the signification of efficient communications there. Besides, he revised our collaborative paper rigidly and even reformatted text fonts and alignments. I got in touch

Acknowledgements

with squash for the first time in weekly team building there and had lots of fun being with him and his team.

I thank all colleagues and collaborators. I thank Prof. Richard Bamler for his insightful and constructive comments on our collaborative paper. I would like to give my special thanks to Eike Hoffmann, Dr. Yilei Shi, Dr. Anja Rösel, and Yingjie Schreiber-Liu for managing IT and secretary issues that ensure and ease my research life. Besides, I thank those who offer assistance, suggestions, and kindness in my Ph.D. period: Pu Jin, Konrad Heidler, Zhenghang Yuan, Dr. Cong Luo, Thomas Stark, Prof. Michael Schmitt, Kun Qian, Lanlan Rao, Kalifou René B. Traore, Dr. Rong Liu, Dr. Yuanyuan Wang, Dr. Chunping Qiu, Dr. Wei Yao, Yuxing Xie, Syed Mohsin Ali, Di Hu, Jianzhe Lin, Xinyi Liu and so on. Besides, I am also deeply grateful to colleagues as well as ESPACE seniors/junior, Dr. Yao Sun, Dr. Jingliang Hu, and Qingyu Li, who support each other in struggling for the doctoral degree. I also appreciate Quanxing Wan for his warm welcome in Wageningen and introducing me to his friends. Truly thanks to new colleagues and friends I met near the end of my Ph.D.: Dr. Qian Song, Yi Wang, Fan Fan, Sining Chen, Chenying Liu, Dr. Xinyi Tong, Dr. Zhitong Xiong, and Fahong Zhang. I really enjoy the time with them spent hiking, wandering, having meals/drinks, and game nights. Besides, recalling those expert-level mountains we have conquered can motivate me when faced with research challenges.

Last but most importantly, I would express my sincere gratitude to my parents and love: Li Hua, Yanping Tang, and Yingya Xu. I do thank my parents for always trusting me and keeping in contact with me even though I seldom contact them actively. Without their solid backup, I could hardly complete my Ph.D. study. Moreover, very much thanks and apologies to Yingya Xu, who married me at Feb. 01, 2021 but stayed apart for most of the time since we met in Wuhan University. I always remember the moments we laugh and cry, the travels we went to Santorini, Paris, Venice, and Tokyo, and the birthday wishes she made. I would especially appreciate her patience and trust and struggle even harder in return.

Contents

Abstract	iii
Zusammenfassung	v
Acknowledgements	vii
List of Abbreviations	xiii
1 Introduction	1
1.1 Motivation and Objectives	1
1.2 Thesis Outline	3
2 A Glance at Deep Learning	5
2.1 Convolutional Neural Networks	5
2.2 Semantic Segmentation Networks	9
3 Aerial Scene Understanding in the Lab	11
3.1 Single-scene Recognition	12
3.1.1 Feature extraction architectures	13
3.1.2 Feature fusion techniques	16
3.2 Multi-label Object Classification	18
3.2.1 binary relevance algorithms	18
3.2.2 Label relation mining algorithms	20
3.3 Semantic Segmentation of Aerial Imagery	21
3.4 Data and Evaluation Metrics	22
3.4.1 Datasets	22
3.4.2 Evaluation metrics	25
4 Aerial Scene Understanding in the Wild	29
4.1 Multi-scene Recognition	29
4.1.1 From Single- to Multi-scene Recognition	29
4.1.2 Deep Learning for Multi-scene Recognition	31
4.2 Semantic Segmentation of Aerial Imagery with Sparse Scribbled Annotations	32
4.2.1 From Dense to Sparse Pixel-wise Annotations	32
4.2.2 Preliminaries	33
4.2.3 Learning with Sparse Scribbled Annotations	35
4.3 Data and Evaluation Metrics	36
5 Summary of works	41
5.1 Exploiting label correlations with bidirectional LSTM for multi-label object classification	41
5.1.1 Motivation	41

5.1.2	Methodology	42
5.1.3	Results	46
5.2	Reasoning about label relations for multi-label object classification	48
5.2.1	Motivation	48
5.2.2	Methodology	48
5.2.3	Results	52
5.3	Memorizing scene prototypes for multi-scene recognition	53
5.3.1	Motivation	53
5.3.2	Methodology	53
5.3.3	Results	55
5.4	A large-scale dataset and benchmark for multi-scene recognition	57
5.4.1	Motivation	57
5.4.2	Benchmark	57
5.4.3	Results	59
5.5	Semantic segmentation of aerial imagery with sparse annotations	62
5.5.1	Motivation	62
5.5.2	Methodology	62
5.5.3	Results	65
6	Conclusion and Outlook	67
6.1	Summary	67
6.2	Outlook	68
	List of Figures	71
	List of Tables	75
	Bibliography	77
A	Yuansheng Hua*, Lichao Mou*, and Xiao Xiang Zhu, “Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional LSTM network for multi-label aerial image classification,” <i>ISPRS Journal of Photogrammetry and Remote Sensing</i> , vol. 149, pp. 188-199, 2019. (* equal contribution)	91
B	Yuansheng Hua, Lichao Mou, and Xiao Xiang Zhu, “Relation network for multilabel aerial image classification,” <i>IEEE Transactions on Geoscience and Remote Sensing</i> , vol. 58, no. 7, pp. 4558-4572, 2020.	125
C	Yuansheng Hua, Lichao Mou, Jianzhe Lin, Konrad Heidler, and Xiao Xiang Zhu, “Aerial scene understanding in the wild: Multi-scene recognition via prototype-based memory networks,” <i>ISPRS Journal of Photogrammetry and Remote Sensing</i> , vol. 177, pp. 89-102, 2021.	141
D	Yuansheng Hua, Lichao Mou, Pu Jin and Xiao Xiang Zhu, “Multi-Scene: A large-scale dataset and benchmark for multiscene recognition in single aerial images,” <i>IEEE Transactions on Geoscience and Remote Sensing</i> , in press, 2021.	179

- E Yuansheng Hua, Diego Marcos, Lichao Mou, Xiao Xiang Zhu, and Devis Tuia, “Semantic segmentation of remote sensing images with sparse annotations,” *IEEE Geoscience and Remote Sensing Letters*, in press, 2021. 195

List of Abbreviations

Abbreviation	Description
3-D	three-dimension.
AA	Average Accuracy.
AI	Artificial Intelligence.
AP	Average Precision.
BoVW	Bag-of-Visual-Words.
CIE	International Commission on Illumination (Commission Internationale de l'Eclairage).
CNN	Convolutional Neural Network.
CORINE	Coordination of Information on the Environment.
CRF	Conditional Random Field.
DCA	Discriminant Correlation Analysis.
DGT	Spanish Directorate General for Road Traffic (Dirección General de Tráfico).
FCN	Fully Convolutional Network.
FN	False Negative.
FN_c	False Negative with respect to the c -th class.
FN_k	False Negative with respect to the k -th example.
FP	False Positive.
FP_c	False Positive with respect to the c -th class.
FP_k	False Positive with respect to the k -th example.
GCN	Graph Convolutional Network.
GGNN	Gated Graph Neural Network.
HED	Holistically-nested Edge Detection.
HOG	Histogram of Oriented Gradients.
IFK	Improved Fisher Kernel.
ILSVRC	ImageNet Large Scale Visual Recognition Challenge.
IoU	Intersection over Union.
LBP	Local Binary Pattern.
LDA	Latent Dirichlet Allocation.

Abbreviation	Description
LIB	Layered Instance Bag.
LSTM	Long Short-Term Memory.
mCF ₁	mean Class-based F ₁ .
mCP	mean Class-based Precision.
mCR	mean Class-based Recall.
mEF ₁	mean Example-based F ₁ .
mEP	mean Example-based Precision.
mER	mean Example-based Recall.
MLP	Multilayer Perceptron.
NAS	Neural Architecture Search.
NMTF	Non-negative Matrix Tri-Factorization.
OA	Overall Accuracy.
OF ₁	Overall F ₁ .
OP	Overall Precision.
OR	Overall Recall.
OSM	OpenStreetMap.
PCA	Principal Component Analysis.
pLSA	probabilistic Latent Semantic Analysis.
ReLU	Rectified Linear Unit.
RF	Random Forest.
RGB	Red, Green, Blue.
RNN	Recurrent Neural Network.
SIB	Segmented Instance Bag.
SIFT	Scale Invariant Feature Transform.
SLIC	Simple Linear Iterative Clustering.
SVM	Support Vector Machine.
the US	the United States.
TP	True Positive.
TP _c	True Positive with respect to the c -th class.
TP _k	True Positive with respect to the k -th example.
UAV	Unmanned Aerial Vehicle.
USGS	the United States Geological Survey.

1 Introduction

1.1 Motivation and Objectives

With the tremendous advancement of earth observation technologies, a considerable volume of remote sensing images is nowadays available and benefits various real-world applications, such as urban mapping, ecological monitoring, urban planning, disaster monitoring, terrain surface analysis, ecological scrutiny, geomorphological analysis, and traffic management. Among all types of images, aerial imagery captured from an aerial perspective is now drawing increasing worldwide attention due to its high spatial resolution and real-time data acquisition. To name a few, Spanish Directorate General for Road Traffic (Dirección General de Tráfico, DGT) introduces aerial photography to traffic monitoring by deploying unmanned aerial vehicles (UAVs) across the country¹. In Qinghai, aerial imaging techniques are exploited to inspect large areas of photovoltaic panels and detect indiscernible panel defects in photovoltaic plants². In January 2021, Cyclone Eloise hit Mozambique, Malawi, Eswatini, Zimbabwe, and South Africa, and during the severe disaster, leveraging aerial imagery of damaged regions helps to search for survivors and save numerous lives from the disaster³.

As a bridge between such successful use cases and aerial imagery, aerial scene understanding, which aims at perceiving and interpreting aerial scenes, has attracted growing research interests during the past decades. In these studies, a *scene* is defined as an association of multiple ground objects (e.g., cars, trees, and buildings) that vary in categories and properties but relate to each other in a certain context, and has a specific thematic meaning (e.g., residential, parking lot, and commercial). In comparison with the terminology *object*, *scene* is a high-level and abstract concept and arduous to determine owing to its high intra-class variation and low inter-class diversity. Therefore, to understand an aerial scene (i.e., a scene taken from the nadir view), a human annotator or visual system should not only identify its compositions but also parse their spatial layouts and underlying correlations. This is effortless for human beings owing to their inherent capabilities of relational reasoning but not easy for machines. Hereby, many efforts have been made to develop intelligent visual recognition algorithms for automatically perceiving aerial scenes. Depending on the level of detail, researches in this field can be sorted into three directions: aerial scene recognition, multi-label object classification, and semantic segmentation of aerial imagery. More specifically, aerial scene recognition often refers to categorizing an aerial image into one of the predefined scene classes and presents a scene-level understanding. Multi-label object classification aims at identifying all objects co-occurring in each aerial image and assigning one image multiple object labels, which presents a macroscopic view of scene compositions. In contrast, semantic segmentation of aerial imagery

¹<https://trans.info/en/spain-dgt-to-use-dozens-of-control-drones-to-spot-driving-infringements-245989>

²<https://guangfu.bjx.com.cn/news/20200818/1098030.shtml>

³<https://www.nepad.org/blog/birds-eye-view-application-of-drone-technology-rapid-disaster-response-management>

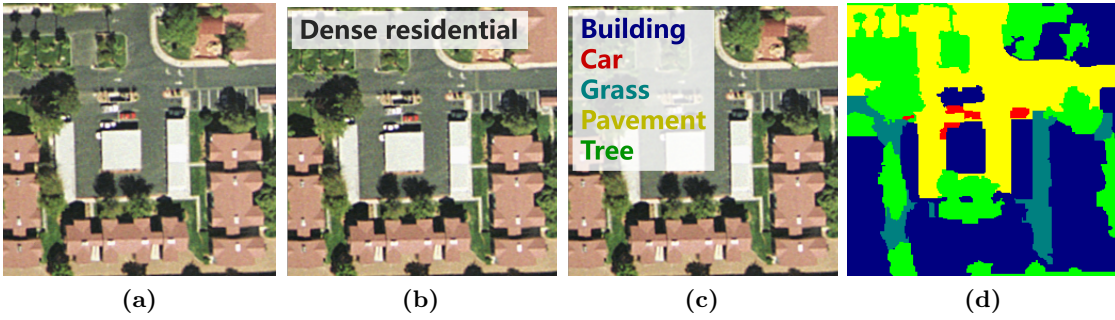


Figure 1.1: Examples of different scene understanding tasks. Given an example aerial image (a), (b) scene recognition aims at predicting the scene category, while (c) multi-label object classification targets at identifying multiple co-existing objects. In (d) semantic segmentation, the category of every pixel should be inferred. The legendary of (d) can be referred to in (c).

interprets an image from the microscopic perspective by classifying the object category of every single pixel. Figure 1.1 shows an example of each scene understanding task. Given an image (Figure 1.1a), aerial scene recognition is to identify its scene label, i.e., *dense residential*, while multi-label object classification targets at inferring labels of all present objects that are *building*, *car*, *grass*, *pavement*, and *tree*. As to semantic segmentation of aerial imagery, each pixel is classified into one of the predefined object types forming a dense semantic mask of the given image.

Although enormous achievements have been obtained in aerial scene understanding under the overwhelming trend of deep learning recently, we observe that most of them impose constraints on experimental prerequisites and study the problem in the laboratory circumstance. As to aerial scene recognition, researches share a common assumption that an aerial image belongs to only one scene and thus assign each image a single label according to its dominant or central scene. In order to comply with this assumption, data producers should acquire and crop aerial images very prudently, ensuring that target scenes are center-aligned in corresponding images or take their majority. Taking Figure 1.1b as an example, the image is labeled as *dense residential* due to that densely connected residential buildings and pavements occupy the majority of the given image. However, such restrictions may trigger unexpected failures, when deploying trained models in the real-world scenario where multiple scenes may co-exist. In addition, we can also observe that there co-exist small-scale parking lots in Figure 1.1b, but they are neglected even *parking lot* is one of the scene categories in the dataset. As a consequence, networks may suffer from confusing supervisory signals in the training process and are prone to learn inappropriate feature representations. Another typical prerequisite in the lab is that annotations required for network training are sufficient and accessible, and this is especially crucial for researches in semantic segmentation of aerial imagery. As shown in Figure 1.1d, dense pixel-wise annotations are essential for semantic segmentation networks to learn to predict the category of each pixel in a given aerial image. However, yielding such annotations is extremely labor- and time-consuming due to the high complexity of aerial image contents, such as irregular land cover boundaries and ambiguous shadowed areas. Therefore, the problem of data insufficiency is quite common in the field of semantic segmentation of aerial imagery, which restricts its deployment in real-world applications.

Towards a practical scenario, this dissertation aims to study aerial scene understanding not only in the lab but also in the wild. To be more specific, we decompose the research topic into four objectives as follows:

- **Understanding aerial scenes from a fine-grained object perspective.**

We observe that aerial scenes have huge intra-class variation, and images belonging to the same scene category can have different objects even in the lab. Hence in comparison with aerial scene recognition, identifying all objects present in an aerial image is essential to offer a more comprehensive view and deliver richer semantic information.

- **Bridging gaps between aerial scene recognition in the lab and wild.**

Aerial scene recognition in the wild is more challenging because images are collected without any constraints, such as centering target scenes and refraining from clutter scenes. Currently, very few efforts have been deployed in this field, and relevant datasets are significantly scarce, which further hinders progress. Nonetheless, we note that there is a vast number of well-annotated single-scene images in the remote sensing community. Thus a question arises naturally that “*can we apply large-scale datasets produced for aerial scene recognition in the lab to that in the wild?*”

- **Data generation for aerial scene recognition in the wild.**

Large-scale well-annotated data is crucial to train deep learning-based algorithms. However, in the remote sensing community, very few datasets consist of unconstrained aerial images, which impedes the advancement of researches in unconstrained aerial scene recognition. Therefore, the community urgently needs large-scale datasets where unconstrained aerial images are captured and assigned multiple labels according to all present scenes.

- **Learning aerial scene parsing models with sparse annotations.**

Dense pixel-wise annotations are difficult to yield, limiting the number of available datasets for semantic segmentation of aerial imagery. Besides, the huge time cost makes it infeasible to learn aerial scene parsing models in real-world applications that need fast responses. To mitigate the heavy annotation burden, learning deep networks with easy-to-draw sparse annotations is now obtaining great research interests, and devising efficient training pipelines will be worth the effort in future research.

1.2 Thesis Outline

This is a cumulative dissertation that reaches the abovementioned four research objectives in the following five peer-reviewed journal papers:

- Yuansheng Hua*, Lichao Mou*, and Xiao Xiang Zhu. Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional LSTM network for multi-label aerial image classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 149, pp. 188-199, 2019. (* equal contribution)

1 Introduction

- Yuansheng Hua, Lichao Mou, and Xiao Xiang Zhu. Relation network for multilabel aerial image classification. *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 7, pp. 4558-4572, 2020.
- Yuansheng Hua, Lichao Mou, Jianzhe Lin, Konrad Heidler, and Xiao Xiang Zhu. Aerial scene understanding in the wild: Multi-scene recognition via prototype-based memory networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 177, pp. 89-102, 2021.
- Yuansheng Hua, Lichao Mou, Pu Jin and Xiao Xiang Zhu. MultiScene: A large-scale dataset and benchmark for multiscene recognition in single aerial images. *IEEE Transactions on Geoscience and Remote Sensing*, in press, 2021.
- Yuansheng Hua, Diego Marcos, Lichao Mou, Xiao Xiang Zhu, and Devis Tuia. Semantic segmentation of remote sensing images with sparse annotations. *IEEE Geoscience and Remote Sensing Letters*, in press, 2021.

The remaining of this cumulative dissertation is organized as follows. Chapter 2 introduces the development of deep learning and takes a glance at Convolutional Neural Network (CNN) and semantic segmentation networks. Chapter 3 draws a picture of researches concerning aerial scene understanding in the laboratory circumstance, while Chapter 4 presents an insight view of aerial scene understanding in the wild. Chapter 5 summarizes our contributions in the five papers, and Chapter 6 concludes the dissertation and presents an outlook of future works.

2 A Glance at Deep Learning

Aerial imagery has high spatial resolutions and can provide richer spatial contextual information of the earth surface, facilitating a more comprehensive view of aerial scenes. However, as a coin has two sides, high resolution aerial images also bring great challenges to image interpretation algorithms, especially in terms of visual feature extraction. Conventional feature extraction methods mainly rely on manually designing feature descriptors, such as Gabor filters [1], Scale Invariant Feature Transform (SIFT) [2], and Local Binary Pattern (LBP) [3], that depict local structures and textures of an image. However, such hand-crafted descriptors can only extract low-level visual attributes and fail to dig out discriminative semantic information. In addition, the efficiency of low-level features depends on prior human knowledge and may not generalize across various types of data. Therefore, a question has been raised: *how to design intelligent systems that can learn to extract high-level features automatically for visual recognition tasks?* As an early attempt, LeCun et al. [4] propose the first but shallow CNN in 1998, and prove its effectiveness in identifying handwritings. With the great progress of computational resources, Alex et al [5] bring the milestone deep CNN, i.e., AlexNet, to the public attention and won the champion of ICLR 2012 that demonstrates its overwhelming performance. Since then, deep learning has been the most popular and dominant solution to visual recognition tasks. Inspired by such great success, deep learning-based algorithms have attracted growing research interests in aerial scene understanding and obtained massive achievements [6, 7, 8].

Therefore, before diving into deep learning scene understanding methods, we briefly review the development of deep learning in this chapter. To be specific, Chapter 2.1 introduces popular CNN architectures, and Chapter 2.2 further delineates the application of CNNs in the pixel-wise understanding of high resolution images.

2.1 Convolutional Neural Networks

CNNs are characterized by hierarchical stacks of convolutional and pooling layers as well as skip connections that facilitate identical mappings. With the increasing depth of CNNs, deeper layers are capable of extracting discriminative semantic features that are proven to be essential for understanding image contents. To have a knowledge of CNNs, we introduce CNN architectures that are popular and often taken as baselines in this section.

LeNet [4]. LeCun et al. first successfully train a handwriting recognition CNN, i.e., LeNet-5, through backpropagation. As shown in Figure 2.1, LeNet comprises three convolutional layers (C1, C3, and C5), two average pooling layers (S2 and S4), and one fully-connected layer (F6). The size of all convolutional filters is 5×5 , and the pooling window of each downsampling layer has a size of 2×2 pixels. In contrast to its following works, not all feature maps produced from S2 are taken as the input of each convolutional filter in C3, and the sigmoid function is utilized to activate outputs of intermediate lay-

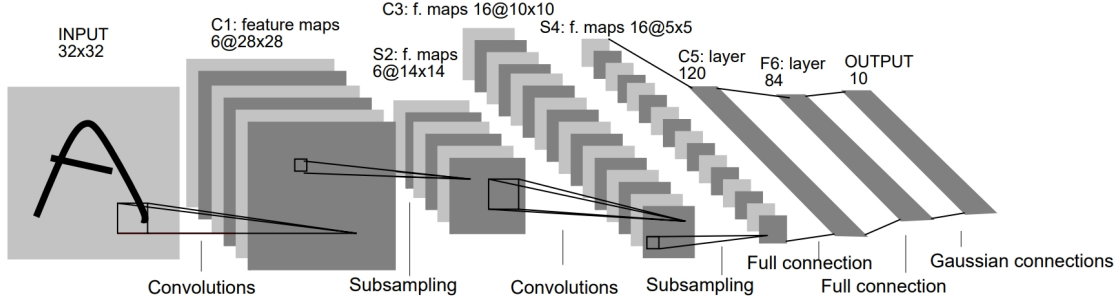


Figure 2.1: The architecture of LeNet-5 [4]. Each plane represents a feature map. $C1$, $C3$, and $C5$ are convolutional layers with 5×5 filters. $S2$ and $S4$ are subsampling layers that halve the width and height of feature maps. $F6$ is a fully-connected layer.

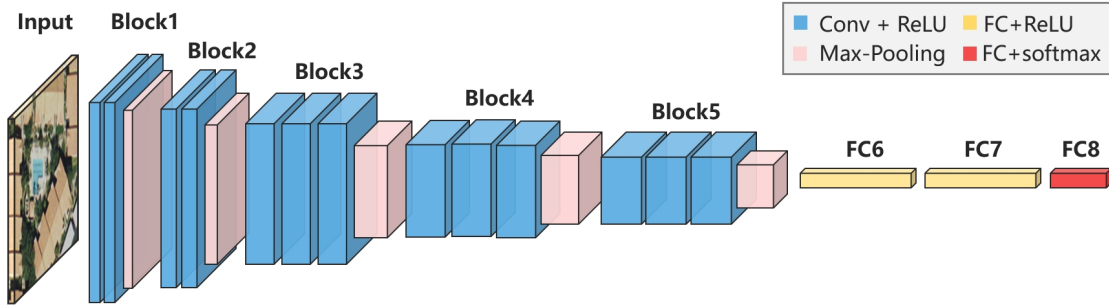


Figure 2.2: The architecture of VGG-16. *Conv* and *FC* stand for convolutional layer and fully-connected layer, respectively.

ers. Experiments are conducted on the MNIST dataset¹, and LeNet-5 surpasses machine learning algorithms that showcases the competence and potential of CNNs.

AlexNet [5]. Thanks to the booming computational efficiency, CNNs are capable of going deeper and handling images with larger scales and more complex contents. In 2012, Krizhevsky et al. propose AlexNet and won the championship of ImageNet Large Scale Visual Recognition Challenge (ILSVRC) by reaching a top-5 error of 15.3%. In contrast to the second-best model, which is built on hand-crafted features [9] and gains a top-5 error of 26.2%, AlexNet learns to automatically extract high-level features through a stack of convolutional and fully-connected layers. Specifically, AlexNet consists of five convolutional layers, where filter sizes vary from 3×3 to 7×7 pixels, and three fully-connected layers. To enlarge the channel dimension of feature maps without increasing computational consumption, max-pooling layers are employed to reduce spatial sizes of features before feeding them to convolutional layers with more filters. As a consequence, the learned high-level feature maps are diverse but at the cost of losing fine-grained spatial information. Besides, instead of the sigmoid function in LeNet, Rectified Linear Unit (ReLU) is selected as the activation function due to its non-saturating non-linearity and less computational complexity. Data augmentation and dropout techniques are introduced to avoid the problem of overfitting during the training phase. To summarize, the great success of AlexNet demonstrates the outstanding performance of deep CNNs and opens a new era for the whole computer vision community.

¹<http://yann.lecun.com/exdb/mnist/>

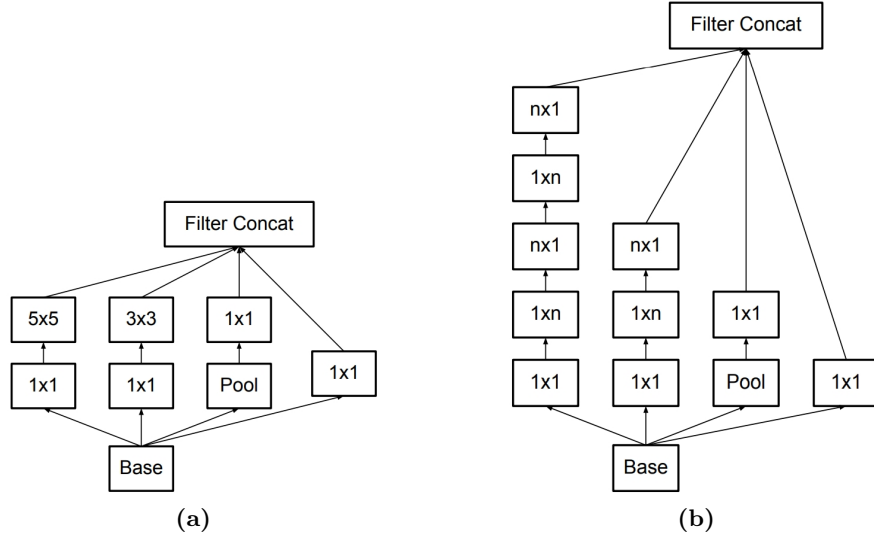


Figure 2.3: Architectures of (a) Inception-v1 and (b) Inception-v3 modules [11].

VGGNet [10]. Simonyan and Zisserman arrange convolutional layers in a block-wise manner and propose VGGNet which is the runner-up at ILSVRC 2014. Specifically, VGGNet is composed of five convolutional blocks and three fully-connected layers, and following each block, one max-pooling layer is attached to downsample feature maps. The size of all convolutional filters is 3×3 , and convolutional layers in each block have the identical number of filters. Taking VGG-16 as an example, each layer in the five blocks has 64, 128, 256, 512, and 512 convolutional filters, respectively. It is noteworthy that compared to directly leveraging large filters, reducing the size of convolutional filters and enlarging the number of layers can 1) ensure comparative receptive fields of deep layers with fewer parameters and 2) enhance the non-linearity of the proposed model. Following this work, designing CNNs block by block manner has been a mainstream trend.

Inception networks [12, 13, 11, 14]. The insight of Inception networks is to perceive images through various receptive fields, and thus, filters with variant sizes are employed to extract feature maps of the same level. Figure 2.3a illustrates the architecture of the Inception module in GoogLeNet (a.k.a. Inception-v1). It can be seen that 1×1 , 3×3 , and 5×5 convolutions and 3×3 max-pooling are conducted on inputs, respectively, and extracted feature maps are concatenated as the final output. Following this design philosophy, Inception-V3 module factorizes a $n \times n$ convolution into $n \times 1$ and $1 \times n$ convolutions (cf. Figure 2.3b) which are more parametrically and computationally efficient. Consequently, Inception networks go not only deeper but also wider, and in 2014, Inception-v1 won the championship of ILSVRC by reaching a top-5 error of 6.67%.

ResNet [15]. He et al. declare that learning direct mappings between images and latent representations may result in the problem of degradation. To this end, the authors propose to learn residual mappings with shortcut connections and successfully push the depth of CNNs towards more than 1000 layers. Considering the computational consumption and model performance, ResNet-50 and ResNet-152 are frequently applied to practical visual missions. In ILSVRC 2015, the authors construct an ensemble of several ResNet variations and won first place by reaching 3.57% top-5 error. Due to the limited page width, we only present the visual illustration of a naive ResNet, i.e., ResNet-34, in Figure 2.4.

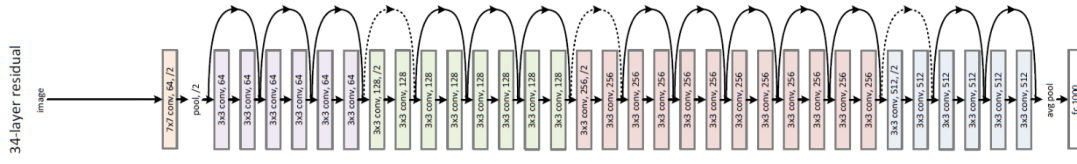


Figure 2.4: The architecture of ResNet-34 [15].

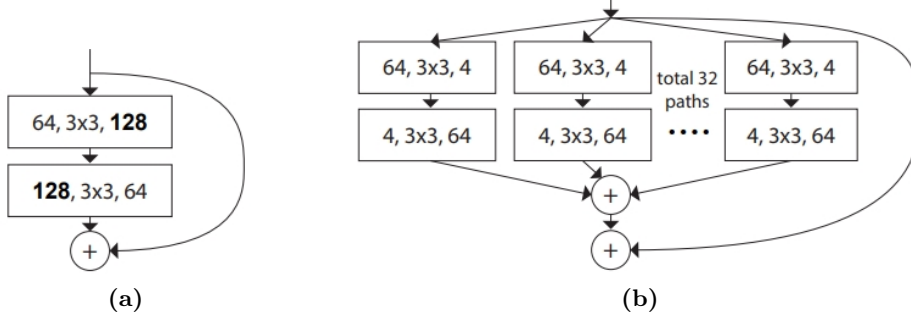


Figure 2.5: Architectures of (a) ResNet and (b) ResNeXt blocks of equivalent complexities [16]. The configuration of each layer is denoted as *the channel dimension of inputs, the size of convolutional filters, and the channel dimension of outputs*.

ResNeXt [16]. ResNeXt is a follow-up to ResNet and won the runner-up in ILSVRC 2016. Specifically, Xie et al. substitute parallel residual transformations (e.g., convolutions) for the traditional stack of convolutional layers, and then aggregate transformed features with element-wise addition. Thus, the entire procedure is so-called aggregated residual transformations and is positioned at the same place as the conventional residual learning. Figure 2.5 compares ResNet and ResNeXt blocks that have the same number of parameters.

DenseNet [17]. DenseNet is proposed to enhance information flow by directly connecting each layer to all subsequent layers with equivalent feature-map sizes. To preserve information learned by proceeding layers, concatenation is employed to combine features from various layers. By reusing feature maps throughout the entire network, DenseNet can learn compact internal representations for visual recognition tasks. Figure 2.6 illustrates the architecture of three consecutive dense blocks.

Light-weight CNNs. Although the booming development of CNNs brings a substantial breakthrough in vision algorithms, training and deploying a deep CNN takes a large amount of computational consumption, which restricts their applications on mobile platforms. Therefore, instead of boosting the network classification capability, another research direction is to preserve the network performance with light loads. As one of representative light-weight CNNs, MobileNet [18] employs depthwise separable convolutions where standard convolutions are factorized into depthwise and pointwise convolutions [19] (cf. Figure 2.7). In its advanced version [20], inverted residual connections and linear bottlenecks are proposed to further unleash the network potential. Moreover, ShuffleNet [21] conducts pointwise convolutions on grouped features separately and rearranges channels of feature maps for facilitating information exchange along the channel dimension. In addition, SqueezeNet [22] improves the computational efficiency by reducing sizes of convolutional filters and reusing low-level features through bypass connections.

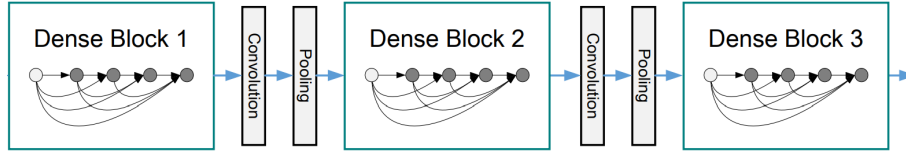


Figure 2.6: The illustration of three consecutive dense blocks [17]. In each block, darker nodes denote higher-level feature maps, while light nodes represent low-level features. Each node is connected to all its subsequent nodes in the common block. Curved arrows denote identity mappings.

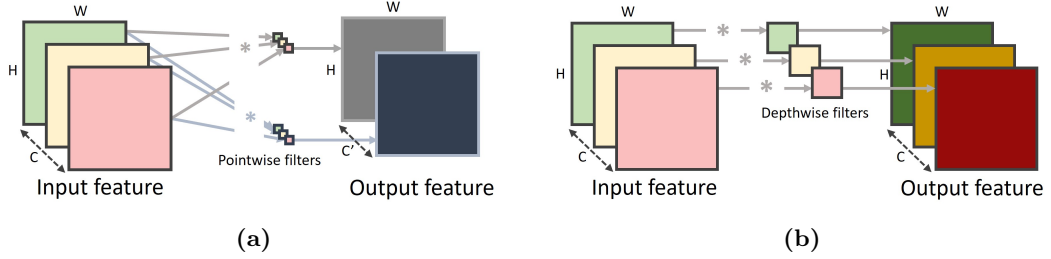


Figure 2.7: Illustration of (a) pointwise and (b) depthwise convolutions. Planes of variant colors indicate different feature channels. Given feature maps with size of $H \times W \times C$ (representing the height, width, and channel), the size of each pointwise convolutional filter is $1 \times 1 \times C$, and that of a depthwise filter is $K \times K \times 1$. Notably, the number of depthwise convolutional filters is required to set as C , and K is arbitrarily defined.

MnasNet [23]. MnasNet architectures are automatically learned on target datasets through the mobile Neural Architecture Search (NAS) technique [23]. Compared to conventional NAS techniques [24], mobile NAS aims to search architectures with low inference latency on mobile platforms. Therefore, MnasNet has a low model latency and achieves a good trade-off between accuracy and latency. To manipulate the model scale, a depth multiplier is committed to shrinking channels of extracted features in each layer. In our experiments, the depth multiplier is set to 1, and the best-performing MnasNet searched on the ImageNet dataset [25] is chosen to perform multi-scene recognition in the wild.

2.2 Semantic Segmentation Networks

Semantic segmentation refers to identifying the category of every pixel in a given image and producing a segmentation mask of the same size as the input image. Under the trend of deep learning, extending CNNs to pixel-wise image interpretation is inevitable and has achieved progress during the past few years. This section gives a glimpse of milestone semantic segmentation networks that are often taken as baselines.

Fully Convolutional Network (FCN) [26]. Long et al. made the first attempt to train semantic segmentation networks adapted from classification CNNs in an end-to-end manner. Specifically, the authors convolutionalize all fully-connected layers by replacing its units with convolutional filters covering equivalent regions. By doing so, FCN can take as input images of arbitrary scales. To enable pixel-to-pixel predictions, high-level feature maps are upsampled with deconvolutions, and low-level feature maps are reused to improve spatial details in generated segmentation masks. Moreover, another benefit of

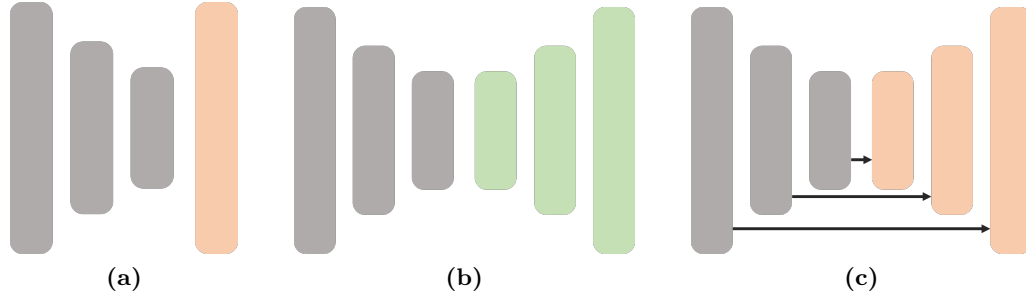


Figure 2.8: Illustration of (a) FCN, (b) SegNet, and (c) U-Net. Gray bars indicate convolutional and pooling layers in the encoder, while orange bars represent deconvolutional and convolutional layers in the decoder. Besides, Green bars denote upsampling and convolutional layers in SegNet. Arrows represent skip connections.

convolutionalizing existing classification CNNs is that their weights pretrained on large-scale image datasets can be employed to initialize corresponding semantic segmentation networks. Figure 2.8a shows the architecture of FCN.

SegNet [27]. SegNet takes VGG-16 as the backbone for extracting high-level features, and then decodes segmentation masks with a mirror architecture of the encoder. In the decoder, max-pooling layers in between convolutional blocks are replaced with upsampling operations. To preserve spatial details and alleviate the learning burden, pooling indices, which record the position of maximum in each local region, are employed to allow non-linear upsampling operations. Figure 2.8 illustrates the difference between SegNet and other early heuristic architectures².

U-Net [28]. U-Net is proposed for biomedical image segmentation, and then becomes popular in semantic segmentation of high resolution natural and aerial images owing to its outstanding capability of boundary delineation. As shown in Figure 2.8c, U-Net thoroughly reuses low-level features by concatenating them with upsampled high-level feature maps. With this design, severe loss of fine-grained structural and textural features can be refrained from to a great extent.

DeepLab networks [29, 30]. Spatial information loss caused by pooling layers is always the trouble in semantic segmentation. To address this, DeepLab [29] proposes to leverage atrous convolutions, which are implemented by dilating convolutional filters with holes (i.e., zeros), and Conditional Random Field (CRF). Compared to standard convolutions, atrous convolutions have larger receptive fields for capturing higher-level features. In DeepLab, convolutions in deep layers are replaced with atrous convolutions, and relevant pooling layers are discarded for remaining spatial resolutions of feature maps. Furthermore, inspired by PSPNet [31], its advanced version [30] employs atrous convolutions, which have variant dilation rates, in a cascade and parallel fashion to extract multi-scale feature representations for semantic segmentation.

²SegNet was proposed in 2015, but finally accepted by TPAMI in 2017.

3 Aerial Scene Understanding in the Lab

With the growing spatial resolutions, aerial imagery can provide richer spatial contextual information of the earth surface and facilitate a more fine-grained view of aerial scenes. Compared to remote sensing images with low spatial resolutions (e.g., hyper- or multi-spectral images), aerial imagery is characterized by the facts that 1) single pixels have few thematic meanings and 2) only Red, Green, Blue (RGB) bands are available leading to very limited spectral cues. Therefore, instead of identifying each pixel individually as interpreting hyper- or multi-spectral cubes, studies in aerial scene understanding deploy more efforts in analyzing spatial patterns and geographic distributions of pixels. This is because a single pixel with abundant spectral information is discernable according to the dictionary of spectral signatures, while an individual pixel with only RGB values delivers no valid semantics. Nonetheless, by grouping pixels in high-resolution aerial imagery in accordance with specific patterns, they can then have meaningful semantics and be assigned thematic classes. For example, the semantic class of a single blue pixel is ambiguous, but a rectangular area comprising purely blue pixels may indicate a swimming pool. Hence, in contrast to hyper- or multi-spectral image interpretation where the category of each pixel should be predicted, aerial scene understanding can be decomposed into two research branches depending on different levels of human perception: image-level and pixel-level understanding. To be more specific, given an aerial image, the former is dedicated to recognizing its scene category, so-called *scene recognition*, or identifying all objects present in the given image, *multi-label object classification*, while the latter often refers to perceiving the category of every pixel, *semantic segmentation*. Notably, the semantic segmentation task is not conflict with our previous delineation that a single pixel is not recognizable, as here we first sense pixels as a whole (e.g., a lake) and then assign the category label to each of them (e.g., each pixel is classified as *lake*). Figure 1.1 presents an example aerial image and its corresponding image-level scene/object labels or dense pixel-wise annotations required for model training.

Regarding scene recognition, our literature review demonstrates that researchers tend to study this problem in a laboratory circumstance and impose constraints that an aerial image is supposed to be composed of or mainly occupied by only one scene. Thus, relevant studies focus on inventing algorithms that assign only one scene label to each aerial image, and to distinguish from researches in interpreting unconstrained aerial images, we term conventional scene classification as *single-scene recognition* in the following depictions. Another presetting in the laboratory is that annotations required for network learning should be fully available, and this is especially vital for training semantic segmentation networks, as they need to predict the category of every single pixel. Figure 1.1d is an example of the pixel-wise annotation, and the size of the mask is the same as that of Figure 1.1a. The value of each pixel indicates the semantic class of the corresponding pixel in the original image. Compared to image-level labels, pixel-wise annotations convey semantic information from the microscopic view but at a high cost.

To summarize, aerial scene understanding in a laboratory circumstance is dedicated to inferring image-level scene/object labels of a well-cropped constrained aerial image

3 Aerial Scene Understanding in the Lab

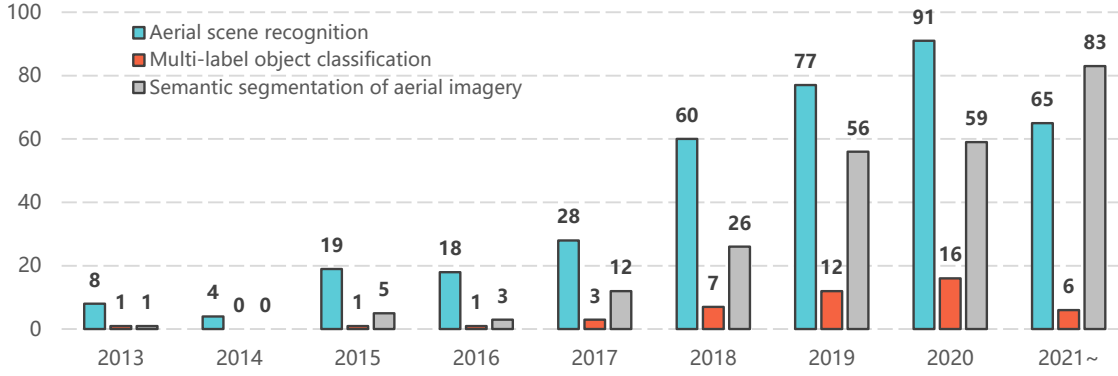


Figure 3.1: Numbers of publications from 2013 till now in aerial scene recognition, multi-label object classification, and semantic segmentation of aerial imagery, respectively. For each task, a search on Web of Science¹ with the query command: $TI=((remote\ sensing\ or\ aerial\ or\ satellite\ or\ land\ use)\ and\ scene\ classification)$, $TI=((remote\ sensing\ or\ aerial\ or\ satellite\ or\ land\ use)\ and\ (multilabel* \ or\ multi-label*))$, and $TI=((remote\ sensing\ or\ aerial\ or\ satellite\ or\ land\ use)\ and\ semantic\ segmentation)$.

or learning to identify each pixel from dense pixel-wise annotations. Although these branches target understanding aerial scenes from variant perspectives, most of the existing researches is built on a common pipeline consisting of two stages: feature learning and decision making. The former can be further decomposed into feature extraction and feature fusion, aiming to extract diverse abstract representations of a given image and aggregate them for the final prediction. The latter is achieved by a classifier that can learn decision boundaries of different classes. In single-scene recognition and multi-label object classification, the classifier has two common implementations: 1) a fully-connected layer followed by an activation function, e.g., the softmax or sigmoid function, 2) a machine learning classifier, e.g., Support Vector Machine (SVM) and Random Forest (RF). In semantic segmentation, a convolutional layer where the number of filters is the same as classes is often taken as the classifier layer. Since most of the literature makes efforts to develop novel feature learning architectures, we hereby sorted them based on their contributions to either feature extraction or feature fusion.

The remaining of this chapter is organized as follows. Chapter 3.1 and 3.2 review recent publications in single-scene recognition and multi-label object classification, respectively. Chapter 3.3 introduces deep learning-based models that are designed for semantic segmentation of aerial imagery and trained in a laboratory scenario.

3.1 Single-scene Recognition

Single-scene recognition refers to identifying the scene present in each image. As a fundamental bridge between aerial imagery and remote sensing applications, single-scene recognition draws huge attention in the community, and massive literatures have been published during the last decades (see blue bars in Figure 3.1). During the literature review, we find that most of the existing studies pay attention to feature learning, and

thus, we provide a detailed review of researches in the development of feature extraction architectures and feature fusion techniques in Chapter 3.1.1 and 3.1.2, respectively.

3.1.1 Feature extraction architectures

The increment of image spatial resolutions brings not only more details about spatial textures and structural patterns of an image but also great challenges to developing feature extraction architectures. In the early years, hand-crafted local and global descriptors, to name a few, SIFT [2], Gabor [1], Histogram of Oriented Gradients (HOG) [32], LBP [3], and color histogram [33], are frequently used to extract local structural patterns and global spectral statistical information. To further construct a holistic representation of an aerial image, low-level visual attributes extracted by these descriptors are then encoded into mid-level representations through approaches, such as Bag-of-Visual-Words (BoVW) [34], Improved Fisher Kernel (IFK) [35], Latent Dirichlet Allocation (LDA) [36], and probabilistic Latent Semantic Analysis (pLSA) [37]. Although early researches [37, 38, 39, 40, 41, 42, 43, 44, 45, 46] have demonstrated their effectiveness, handcrafted feature descriptors may still suffer from limitations, such as poor generalization capability and high dependence of human expert knowledge.

In recent years, the emergence of deep learning has made a breakthrough in single-scene recognition, and many achievements [6, 47, 46] have been attained in this field. By literature review, we observe that deep learning-based algorithms share a common design that they take deep CNNs as the feature extraction backbone and propose task- or data-specific adaptations to boost classification capabilities of invented networks. In [46], the authors conducted an overall comparison of algorithms built on hand-crafted feature engineering and deep neural networks, and quantitative results on various aerial scene datasets demonstrate the superior performance of deep CNNs. The reason is that deep CNNs are composed of hierarchically stacked convolutional layers and can be trained to automatically extract discriminative semantic features via back-propagation. In [48], the authors employ CAM [49] to visualize regions that a CNN pays more attention to when recognizing aerial scenes (cf. Figure 3.2). Specifically, the authors take aerial images of *intersection* and *beach* as examples and visualize CAM learned by VGG-16. Figure 3.2b and 3.2e present regions that a shallow convolutional layer highlights, and Figure 3.2c and 3.2f illustrate that a deep layer identifies *intersection* and *beach* by focusing on the road crossing and wave, respectively. As we can see, convolutional filters in shallow layers perform as hand-crafted local descriptors and extract low-level visual attributes, such as edges and blobs, while deep layers can learn discriminative high-level features (see red regions in Figure 3.2c and 3.2f).

Therefore, to improve the classification performance of deep neural networks, many efforts have been made to develop efficient feature extraction architectures. Early attempts [50, 51, 52, 53, 54] treat CNNs pretrained on large-scale natural image datasets (e.g., ImageNet [25]) as feature descriptors due to that spectral properties of aerial and natural images are similar. To achieve so, researchers discard layers including and after the last fully-connected layer of a CNN and treat generated high-dimensional feature vectors as image representations. However, since imaging techniques of natural and aerial images are different, ground targets present in these two types of images show diverse spatial patterns. Consequently, directly applying a CNN developed for natural images to recog-

¹<https://www.webofscience.com/wos/woscc/advanced-search>

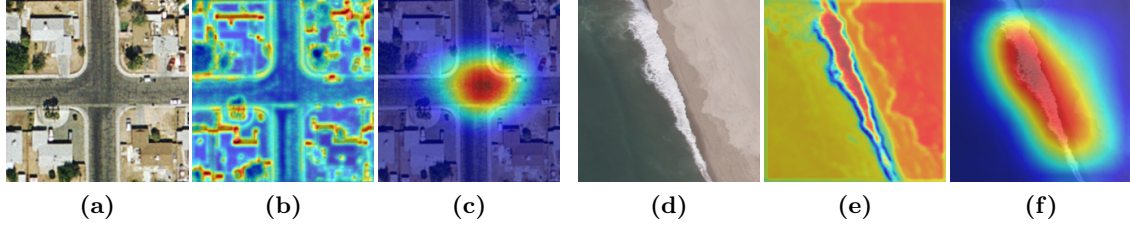


Figure 3.2: Visualization of attentional regions captured by VGG-16 in recognizing (a) *intersection* and (d) *beach*. (b) and (e) are CAMs generated by shallow convolutional layers of VGG-16, while (c) and (f) are CAMs extracted by deep layers.

nizing aerial images imposes constraints on the potential of deep learning in this field. To address this issue, recent studies aim to develop efficient feature extraction architectures for extracting data- and task-specific features. Specifically, there are three main design philosophies:

- Multi-scale feature extraction.** Since the scale of one scene may vary substantially depending on its circumstance and the flying height of airborne platforms, it is fundamental to extract features at different scales by downsampling images or applying convolutions with variant fields of receptive fields (cf. Figure 3.3 (a)). In [55, 52, 56, 57], the authors propose to learn multi-scale features by downsampling an image into variant scales before feeding them to CNNs. Instead of downsampling original images, Tombe and Viriri [58] downsample feature maps with sliding pooling windows and feed them into multi-grained Cascade forests for identifying scene categories. In [59, 60], the authors downsample feature maps of different levels and feed them to independent feature extraction branches. In addition to varying the spatial dimension straightforwardly, convolutions of variant sizes can also extract multi-scale features. In [61, 62], the authors extract multi-scale features by leveraging convolutional filters of variant sizes.
- Multi-level feature extraction.** As shown in Figure 3.2, although high-level features are abstract and discriminative, they lose local spatial patterns which are remained in low-level features. Therefore, the idea of jointly leveraging low- and high-level features for single-scene classification arises naturally. A common way to extract multi-level features is to reuse outputs of shallow layers in a deep neural network (cf. Figure 3.3 (b)). Specifically, Lu et al. [63] exploit features extracted by the last three convolutional blocks of VGG-16 and concatenate such multi-level features and outputs of the penultimate fully-connected layer for the final prediction. In [64], the authors take the first four residual blocks of ResNet50 to extract multi-level features, while in [65], the last three blocks of ResNeXt50 is leveraged. Mei et al. [66] experiment with AlexNet, VGG-19, and ResNet50, and build a dictionary of multi-level features for scene classification. Similarly, Hu et al. [67] divide features of variant levels into multiple feature sets for subsequent ensemble learning. In [68], the authors employ the second, third, and fourth blocks of ResNet-50 and DenseNet-121 to learn low-, middle-, and high-level feature maps, respectively. Sun et al. [69] utilize CNNs and hand-crafted feature descriptors to extract multi-level features.

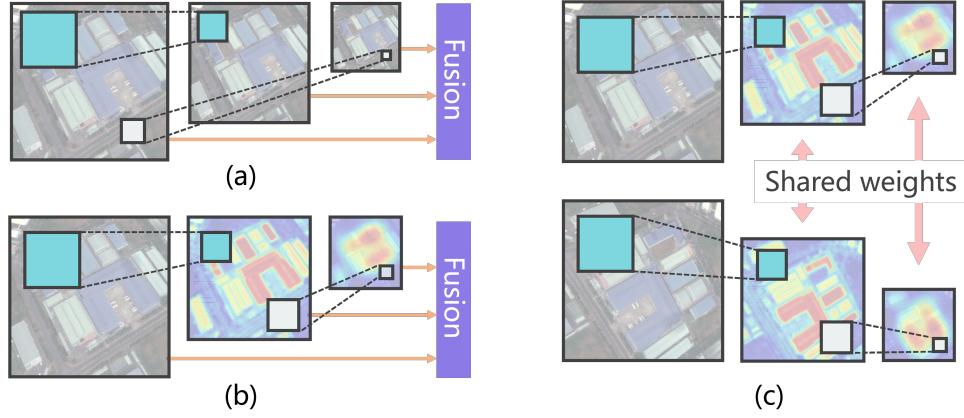


Figure 3.3: Illustration of (a) multi-scale feature extraction, (b) multi-level feature extraction, and (c) rotation-equivariant feature extraction. Notably, in (a), pooling/convolution is conducted in sliding windows, and both images and feature maps can be taken as input.

- **rotation-equivariant feature extraction.** In aerial imagery, scenes are taken from a nadir and thus should be equivariant to rotations. With this intention, Siamese feature extraction architectures are introduced to single-scene recognition aiming to learn rotation-equivariant features. In [70, 71, 72, 73], the authors feed an original aerial image and its rotated version to a Siamese architecture (cf. Figure 3.3 (c)) for jointly learning features from both non-rotated and rotated images. Normally, each branch is an individual CNN and share weights with the other branch. In [72], the authors ensemble individual CNNs learned from images augmented by variant rotations. Xie et al. [74] design a network to not only predict scene categories but also rotated angles.

In addition to network design, other researches [75, 76, 77] aim to improve the efficiency of feature extraction architectures from the perspective of training data. Guo et al. [75] augment training samples by generating pseudo images with GANs. Liu and Ma [76] propose to learn class-wise domain invariant features from different aerial image datasets via adversarial training. Li et al. [77] divides target dataset into variant subsets for training networks to automatically correct uncertain labels. Liu et al. [78] propose to build a hierarchical category tree of scene categories for network training. In [79], the authors resort to object labels of aerial images, which present a microscopic view of scenes, and Luo et al. [80] take label ambiguity into consideration by transforming a single label into a neighbor-based distribution. Zhu et al. [81] introduce GANs to improving spatial resolutions of remote sensing images for single-scene classification. In [82, 83], the authors inject adversarial training samples for improving the robustness of extracted features. Besides, the authors in [84, 85] ensemble different deep neural networks for feature extraction, and Cheng et al. [86] explicitly regularize the network training by imposing a metric learning term on the penultimate fully-connected layer. In [87], the authors investigate representing samples in the non-Euclidean space and introduce the Lie group manifold to scene classification. In addition, we note that an increasing number of researches [88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101] study feature extraction in the scenario of insufficient training samples and limited computational resources. Instead

of heuristic network design, NAS technique is employed to search for optimal network architectures automatically, and achievements [102, 103, 104, 105] have been obtained in single-scene recognition.

3.1.2 Feature fusion techniques

As a bridge between feature extraction and the final inference, feature fusion refers to rearranging and aggregating diverse features for yielding a more discriminative and holistic feature representation. A naive way to fuse features is to uniform their spatial sizes through pooling [65, 63] or reshaping [64] operations and then concatenate them forming high-dimensional feature maps/vectors. However, such a straightforward fusion technique may suffer from redundant features and dilute discriminative features. To this end, massive efforts have been made to derive the importance of each feature map/vector in both unsupervised and supervised manners that are introduced as follows:

- **Unsupervised feature fusion.** To measure the feature redundancy and retain only principal features, it is natural to compute similarities/variances among features and select prominent ones through dimensionality reduction methods, such as Principal Component Analysis (PCA) and Discriminant Correlation Analysis (DCA). Specifically, Li et al. [52] employ IFK and PCA to fuse outputs of convolutional blocks and fully-connected layers, while in [51], the authors adopt DCA instead. Moreover, He et al. [106] propose to calculate the covariance between each two feature maps and project the covariance matrix into Euclidean space with the matrix logarithm operation for inferring scene categories. Yuan et al. [54] regard the feature vector at the image center as the representative feature and measure cosine similarity between features located at other positions and the center one. Those with high relevance are then retained for further fusion. Similarly, Dan and Li [107] excavate the relationship between features of different positions with a matrix outer product.
- **Supervised feature fusion.** Under the booming trend of deep learning, enabling networks to adaptively derive the importance of each feature map/vector is now drawing increasing attention. Related studies mainly resort to spatial and channel attention modules [108] and self-attention mechanism [109]. As to the former, a convolutional layer with one filter is employed to learn spatial attention maps, and channel attentions are captured by jointly utilizing global pooling and fully connected layers. Regarding later, feature maps are transformed to key, query, and value for inferring attention maps where each entity represents the relevance between feature vectors located at two different positions. Figure 3.4 exhibits an visual illustration of them. In [110, 111, 112, 113], the authors develop networks based on spatial and channel attention modules. In [114] self-attention mechanism is introduced to automatically measuring relevances among features from different convolutional layers for feature fusion. In [115], the authors enhance responses corresponding to objects in feature maps via a context-aware spatial attention module. Fu et al. [116] enhance the feature rearrangement by recurrently feeding features into self-attention modules. Qi et al. [117] design an adaptive object-centric pooling operation that emphasizes regions, including objects for classification. Ma et al. [118] design a network to learn the weightings of multi-level features and compute the weighted sum of them as the holistic image representation. To fuse features in the decision level, Shen

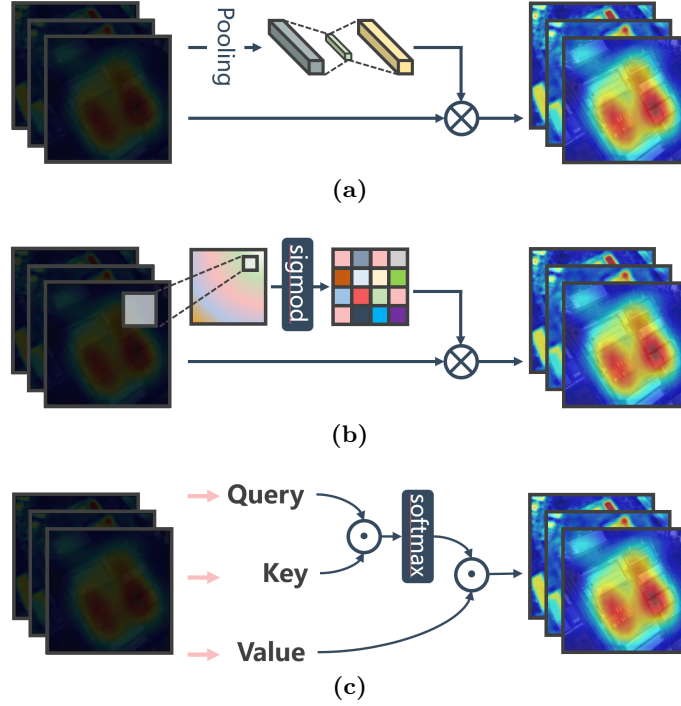


Figure 3.4: Illustration of (a) channel and (b) spatial attention modules, and (c) self-attention mechanism. In (a), global average/max pooling is first conducted, and output feature vectors are fed to fully-connected layers for learning channel attentions. In (b), a spatial attention mask is learned with 1×1 convolutions. To apply (c), feature maps are reshaped along the spatial dimension, which yields a sequence of feature vectors, before they are linearly transformed to query, key, and value.

et al. [119] train multiple classification heads on features of variant levels and fuse them for the final decision. Sun et al. [120] propose to aggregate features of variant levels through a gated bidirectional connection which makes it possible for networks to learn to discard redundant features automatically. Besides, Wang et al. [121] employ Long Short-Term Memory (LSTM) to recurrently learn to localize attentional regions of an aerial image for recognizing its category, and Tong et al. [122] employ a spatial transformer network to adaptively extract discriminative regions and discard redundant features. In [64], the authors propose to preserve spatial textural information contained in features of variant levels with capsule networks, and the authors in [123] aggregate multi-level features into capsule representations for scene categorization.

As a consequence, the fused features are supposed to be more discriminative and concise. Afterwards, to derive scene categories from fused features, a fully-connected layer followed by a softmax function is most commonly used and allows training networks in an end-to-end manner [110, 122, 111, 113]. Besides, several works make use of machine learning classifiers, such as SVM [52, 69] and RF [58], to infer scene labels. By learning holistic feature representations of aerial images, single-scene classification algorithms can make judicious predictions compared to conventional methods that rely on hand-crafted feature engineering.

3.2 Multi-label Object Classification

Multi-label object classification provides an object-level understanding of aerial scenes by identifying categories of objects present in a given scene, and this research topic draws increasing attention in the remote sensing community during the last decades (see red bars in Figure 3.1). To predict multiple object labels, an intuitive scheme is to transform the problem into multiple binary classification tasks, where a binary classifier is trained independently for each label. Algorithms based on this scheme are also known as *binary relevance* algorithms [124, 125]. However, these algorithms suffer from two serve limitations: 1) classifiers with respect to rare object categories are prone to have poor performance due to the deficiency of training samples, and 2) label correlations are not learned and exploited in categorizing coexisting objects. To tackle these limitations, recent studies deploy more efforts to encode label correlations for multi-label object classification, and thus sorted as *label relation mining* approaches. In this section, we introduce these two technique branches in Chapter 3.2.1 and Chapter 3.2.2, respectively.

3.2.1 binary relevance algorithms

Binary relevance algorithms decompose the multi-label classification task into multiple binary classifications. A common design philosophy is that a feature learning module is first employed to extract discriminative image representations which are then fed to multiple independent classifiers with respect to candidate object categories. Figure 3.5 presents a visual illustration of a binary relevance algorithm. Notably, we also consider deep neural networks end with a fully connected layer and a sigmoid activation function as binary relevance algorithms. This is attributed to the property of the sigmoid function where outputs are calculated independently with the following equation:

$$y_k = \sigma(\mathbf{W}_k \mathbf{X}), \quad (3.1)$$

where y_k is the prediction of class k , \mathbf{W}_k denotes weights of the k -th unit, and \mathbf{X} represents feature presentations of the image. Eq. 3.1 suggests that each output is computed independently, and underlying correlations among are thus cut off. Compared to single-scene recognition, the main difference lies only in the last activation layer, which makes it feasible to transfer most of the deep neural networks reviewed in Chapter 3.1 to multi-label object classification by simply replacing the last softmax function with the sigmoid function.

In [125], the authors propose a 3-layer autoencoder to reconstruct input images and feed the learned latent features to a multilayer perceptron activated by a sigmoid function for predicting multiple labels. Following this work, Zeggada et al. [126] introduce deep convolutional neural networks to identifying multiple objects in UAV images and substitute the last softmax layer with a radial basis function neural network (RBFNN) for the final prediction. In RBFNN, the Otsu thresholding algorithm [127] is leveraged to estimate the threshold of deciding whether predicted results indicate the presence or absence. In [124], the authors employ a deep neural network to learn image representations and infer each object label independently with XGBoost [128]. Zegeye and Demir [129] introduce active learning to train confident classifiers by punishing predictions falling inside margins of a classifier. Besides, Bashmal et al. [130] introduce Transformer [109] to multi-label object classification and propose a two-branch architecture for jointly recognizing objects in an image and its augmented version (e.g., being flipped, rotated, and randomly cropped).

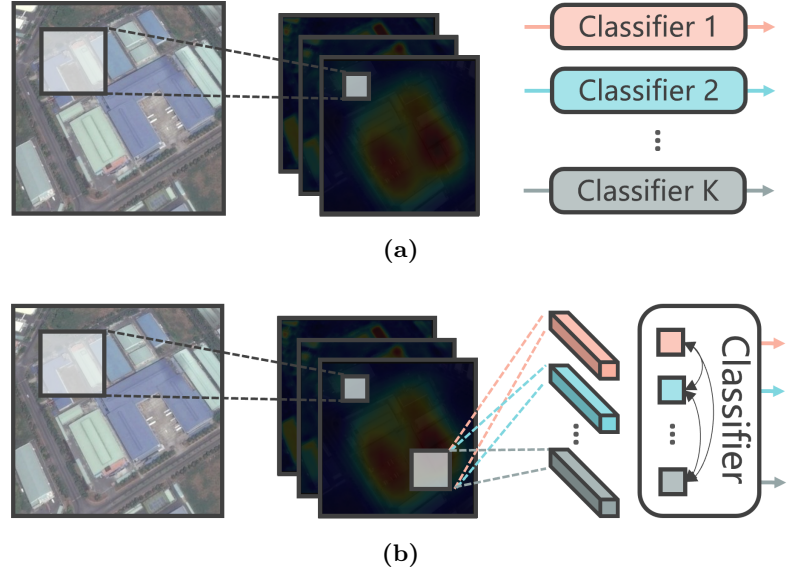


Figure 3.5: Illustration of (a) binary relevance and (b) label relation mining approaches.

In [131], the authors note that aerial images collected from adjacent locations share similar or complementary spatial patterns and are expected to be assigned with comparable labels. Hence, they introduce CRF to encode spatial correlations for refining multi-label predictions. Similarly, Chaudhuri et al. [132] construct a neighborhood graph where each node represents an image, and the weight of each edge is determined by the similarity between corresponding images. Following this work, Dai et al. [133] includes not only local descriptors but also histogram-based spectral descriptors for jointly encoding spatial and spectral characteristics of input images. Zhu et al. [134] propose to jointly exploit scene and object labels for enabling networks to learn discriminative feature representations. Specifically, the proposed network consists of two branches, where one learns to predict multiple object labels, while the other one encourages features of images belonging to the same scene to be closer in the same embedding space. In [135], the author designs a neural network consisting of two branches: an image branch and a label branch. The former is responsible for extracting high-level features and ranking predictions through ResNet50, while the latter learns to match image features and their corresponding ground-truth labels. Furthermore, the authors evaluate the network performance with EfficientNet-B0 as the backbone on the UC-Merced multi-label dataset in [136]. Sumbul et al. [137] exploit a multi-branch architecture to extract semantic information of images with variant spatial resolutions. Afterwards, these features are fed to a bidirectional LSTM for encoding their spatial relations and inferring the existence of each object. Shendryk et al. [138] investigate the influence of different sizes of input images and mini-batches in training a VGG-like model. Instead of directly learning multiple image-level labels, Shao et al. [139] trains a semantic segmentation network to predict pixel-level labels and then induce image-level multiple object labels from segmentation maps. As an interesting trial, Topcu et al. [140] explore the feasibility of capsule networks and validate their performance on the UC-Merced multi-label dataset.



Figure 3.6: Example high resolution aerial images delineating (a) *industrial*, (b) *residential*, and (c) *parking lot* but sharing common object labels, *car* and *pavement*.

3.2.2 Label relation mining algorithms

The mutual dependence among coexisting objects is inherent in multi-label aerial images. Figure 3.6 shows an example that images belonging to different scenes but have several common object labels. Therefore, mining underlying label correlations is crucial for multi-label object classification, which is also in line with human cognition of the world. For instance, a car’s presence highly indicates the co-existence of pavements, and the predicted occurrence of a ship often suggests that there is water around. Label relation mining algorithms [141, 142, 143, 144] can be summarized with the following equation:

$$y_k = \mathcal{F}(\mathbf{X}_k, \mathcal{X}_{\neg k}), \quad (3.2)$$

where \mathbf{X}_k is the feature extracted for k -th object label, and $\mathcal{X}_{\neg k}$ denotes a subset of features with respect to other labels. \mathcal{F} indicates label relation mining functions that can be implemented with LSTM [141], relation networks [142], and graph neural networks [143, 144]. Compared to Eq 3.2, features of non-target labels are taken as input for encoding label relevances.

More specifically, Hua et al. [141] boil the multi-label object classification task down into a structured output problem, where the prediction of each label is dependent on the others. To achieve so, the authors extract class-wise feature representations with a CNN and make use of a bidirectional LSTM to predict the presence of each object at corresponding time steps. Following this work, Ji et al. [145] conduct spatial normalization on class-wise feature maps and predict all classes at each time step. The final result is then generated by max-pooling predictions at all time steps. Diao et al. [146] propose to replace convolutions in CNNs with deformable convolutions for learning geometry-invariant label-related features and then encode dependencies among extracted features using a Gated Graph Neural Network (GGNN). Huang et al. [147] extract label correlation cues for multi-label object classification by computing label co-occurrence matrices from target datasets. However, the applicability of this method is dependent on the prior knowledge of label statistics which is often not available in the real-world scenario. In [144], the authors aim to learn an embedding space where images with similar contents are clustered, and dissimilar images are far away from each other. To exploit image similarities, the authors construct a graph based on the number of labels shared by each image pair and design

a scalable neighbor discriminative loss for training networks. Zhang et al. [148] formulate the multi-label object classification task as a recommendation problem and exploit Non-negative Matrix Tri-Factorization (NMTF) to recover image-label, image-feature, and feature-label matrices. In [149], the authors make use of low-rank representation for constructing feature- and label-based graphs and classify unlabeled images by measuring their semantic similarities with labeled images. Besides, Hua et al. [150] observe that multiple object labels are difficult to create, and noise is inevitable. Therefore, the authors leverage label correlations extracted from pre-trained word embeddings to correct predictions.

Instead of directly predicting multiple labels from the whole image, researches [151, 143, 152, 153] regard an image as an integration of multiple sub-regions which include different objects. Specifically, Chen et al. [151] treat an image as a bag of several instances and thus segment images at different levels of details resulting in multiple instances. Followingly, a hierarchical semantic structure is developed for inferring labels of each segment, which are eventually aggregated as the final prediction. In [143], the authors segment an image into several disjoint regions with parametric kernel graph cuts [154] and extract their features with local descriptors. Afterwards, features are fed to a Graph Convolutional Network (GCN), where each entity of an adjacency matrix is computed by the distance and orientation angle between two region centroids, for inferring multiple labels. Li et al. [152] partition feature maps into superpixels and then build graphs on them for performing multi-label object classification. In [153], the authors utilize a regular grid to partition each image into several patches and decompose the multi-label object classification task into the problem of categorizing the single label of each instance. Segmented Instance Bag (SIB) and Layered Instance Bag (LIB) are used to extract features of each instance, and the Mahalanobis distance-based K-Medoids approach is used to predict labels. By encoding and exploiting label correlations, algorithms are expected to make more prudent decisions of co-occurred object labels.

3.3 Semantic Segmentation of Aerial Imagery

The mainstream pipeline of semantic segmentation networks is identical to classification networks but emphasizes more on reusing low-level features in encoder/decoder subnetworks and pixel-wise feature aggregation. Another difference is that the final classifier in decoder subnetworks is often implemented as a convolutional layer where the size and number of filters are 1×1 and the number classes, respectively. Hence, instead of flattening and global pooling operations that aggregate features of all pixels, concatenation and element-wise addition are more often employed to merge features in a pixel-to-pixel manner.

Baseline segmentation networks are delineated in Chapter 2.2, and Figure 2.8 illustrates most frequent network architectures. Following those milestones, most efforts [155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165] in semantic segmentation of aerial imagery are deployed to enhance the feature representation capability through compact inter-layer information flow. Specifically, Ding et al. [155] propose to inject semantic features extracted by deep layers into low-level features with a channel attention mechanism (cf. Figure 3.4a), so that fine-grained spatial details as well as discriminative semantic information can be jointly used to predict segmentation masks. Li et al. [156] propose lightweight spatial and channel attention modules to adaptively refine features along spatial and channel dimensions. In [160], the authors aim to design a parametric-

efficient semantic segmentation network by reducing the number of parameters as well as resizing and fusing multi-scale features in decoder subnetworks. Marcos et al. [166] propose rotation-equivariant convolutional filters to ensure the extracted features are encoded with high rotation equivariance. In [167], the authors propose to learn multi-scale attentions and fuse multi-level features via self-attention mechanism (see Figure 3.4c) for boosting the network performance in detecting size-varied objects.

Instead of designing elegant network architectures, another research branch is to learn networks in a multitasking manner. In [168], the authors propose a multitask learning strategy where networks are trained to predict not only segmentation masks but also inter-class boundaries. By doing so, networks are expected to learn more discriminative semantic features. Similarly, the authors in [169] employ morphological operations to detect edges and take them as supervisory signals in training networks to learn accurate edges. Diakogiannis et al. [165] propose a multitasking Res-UNet-a, where residual blocks are stacked and connected under the framework of UNet, to infer semantic classes and boundaries as well as reconstruct distance map (distances between pixels and their nearest boundaries) and original images. Besides, the authors propose a Tanimoto loss which is built on the Dice loss but can accelerate the convergence. Li et al. [170] propose networks to learn domain- and rotation-invariant features by imposing three weakly-supervised constraints: weakly-supervised transfer invariant constraint, weakly-supervised pseudo-label constraint, and weakly-supervised rotation consistency constraint. Li et al. [171] introduce self-supervised representation learning to semantic segmentation and design three pretext tasks, i.e., image inpainting, transform prediction, and contrastive learning, for pretraining backbone networks. Experimental results show that networks built on pre-trained backbone networks can achieve satisfactory performance with limited pixel-wise annotations.

3.4 Data and Evaluation Metrics

Large amounts of available training data play a key role in learning deep neural networks for aerial scene understanding. Thanks to the great development of remote sensing techniques, an increasing number of aerial imagery is now available for producing datasets. Table 3.1 presents an overview of datasets published for aerial scene understanding in the laboratory circumstance. In this chapter, we introduce datasets and evaluation metrics in Chapter 3.4.1 and 3.4.2, respectively.

3.4.1 Datasets

The number of aerial image datasets for single-scene recognition has been growing since the UC-Merced dataset [34] is published. To yield single-scene aerial image datasets, producers first crop images from an extremely large-scale image (such as the United States Geological Survey (USGS) National Map, Google Earth imagery, Bing Map and Tianditu imagery),

²<https://map.tianditu.gov.cn/>

³<https://land.copernicus.eu/user-corner/technical-library/corine-land-cover-nomenclature-guidelines>

⁴<http://www.classic.grss-ieee.org/community/technical-committees/data-fusion/2015-ieee-grss-data-fusion-contest/>

⁵<https://www2.isprs.org/commissions/comm2/wg4/benchmark/2d-sem-label-vaihingen/>

⁶<https://www2.isprs.org/commissions/comm2/wg4/benchmark/2d-sem-label-potsdam/>

Table 3.1: An overview of existing public single-scene aerial image datasets.

Dataset	# images	resolution	# scenes	size	source	Year
UC-Merced [34]	2,100	0.3 m/px	21	256×256 px	USGS	2010
WHU-RS19 [172]	1,005	≥ 0.5 m/pix	19	600×600 px	GE	2012
WHU20 [45]	5,000	0.3-7.4 m/px	20	600×600 px	GE	2015
RSSCN7 [173]	2,800	0.2-1.4 m/px	7	400×400 px	GE	2015
SIRI-WHU [174]	2,400	2 m/px	12	200×200 px	GE	2016
RSC11 [175]	1,232	~ 0.2 m/px	11	512×200 px	GE	2017
AID [46]	10,000	0.5-8 m/px	30	600×600 px	GE	2017
NWPU-RESISC45 [176]	31,500	0.2-30 m/px	45	256×256 px	GE	2017
RSI-CB256 [177]	24,000	0.3-3 m/px	35	256×256 px	GE/BM	2017
RSI-CB128 [177]	36,000	0.3-3 m/px	45	128×128 px	GE/BM	2017
RSD46-WHU [178, 179]	117,000	0.5-2 m/px	46	256×256 px	GE/T	2017
AID++ [180]	400,000	0.5-8 m/px	46	600×600 px	GE	2018
PatternNet [181]	30,400	0.06-4.7 m/px	38	256×256 px	GE	2018
OPTIMAL-31 [182]	1,860	-	31	256×256 px	GE	2019
CLRS [183]	15,000	0.26-8.9 m/px	25	256×256 px	GE/BM/T	2020
MLRSN [184]	109,161	0.1-10 m/px	46	256×256 px	GE	2020
So2Sat LCZ42 [185]	400,673	10 m/px	17	32×32 px	S1/S2	2020
Million-AID [47]	$>1,000,000$	0.5-153 m/px	51	600×600 px	GE	2020

GE, BM, T, and S1/S2 denote Google Earth, Bing Map, Tianditu², Sentinel-1, and Sentinel-2 imagery. px indicates pixel(s).

Table 3.2: An overview of existing public multi-label aerial image datasets.

Dataset	# images	resolution	# scenes	size	Label	Year
UCM-mul [132]	2,100	0.3 m/px	17	256×256 px	M	2018
DFC15-mul [141]	3,342	0.5 m/px	8	600×600 px	DFC15	2019
BigEarthNet[186]	590,326	10-60 m/px	43	$\leq 120 \times 120$ px	CLC	2019
AID-mul [142]	3,000	0.5-8 m/pix	17	600×600 px	M	2020
MLRSNet [184]	109,161	0.1-10 m/px	46	256×256 px	M	2020

CLC is the abbreviation of Coordination of Information on the Environment (CORINE) Land Cover database³.

DFC15 indicates the GRSS.DFC.2015⁴ dataset published for 2015 IEEE GRSS Data Fusion Contest.

M denotes annotations are manually yield through visual inspection.

and then manually inspect their contents or resort to crowdsourced data for annotation. A brief summary of existing public single-scene aerial image datasets is presented in Table 3.1. Figure 3.7 shows several examples from variant sources.

Compared to single-scene aerial image datasets, multi-label aerial image datasets are more arduous to yield. This is because annotators are required to visually inspect every one of the objects present in each aerial image for determining its multiple object labels. Although efforts [180, 47] have been made to alleviate such annotation burden by resorting to crowdsourcing platforms, their attempts demonstrate that human labor is still necessary due to the incorrectness and incompleteness of crowdsourced data. Thus, another solution arises that is reproducing from existing pixel-wise annotated databases, such as

3 Aerial Scene Understanding in the Lab

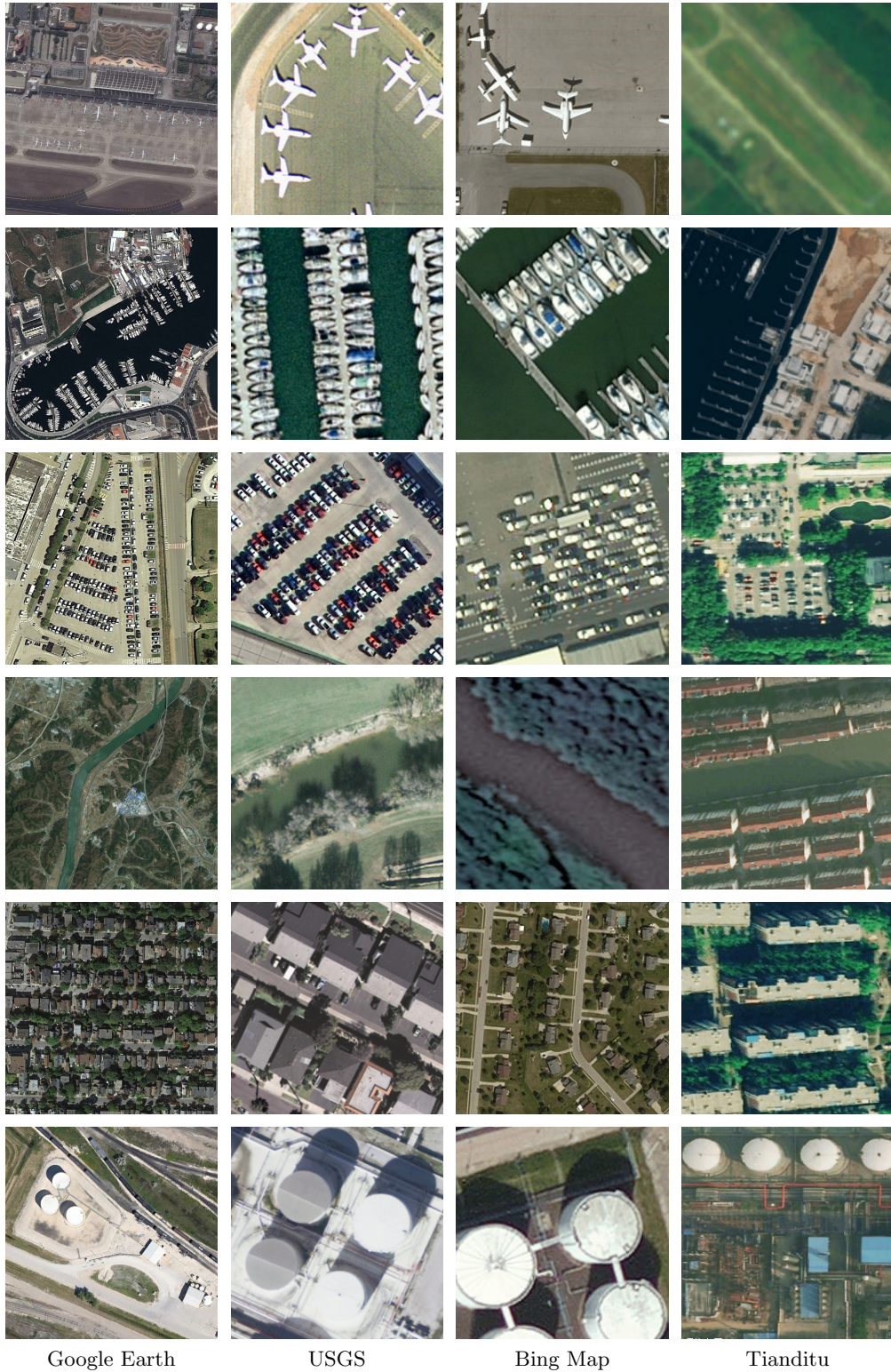


Figure 3.7: Example aerial images from variant datasets (from left to right: AID, UCM, RSI-CB256, and RSD46-WHU) with respect to different scenes (From top to bottom: *airport*, *harbor*, *parking lot*, *river*, *residential*, and *storage tanks*). Data source platforms are denoted in the bottom row.

Table 3.3: An overview of existing high resolution image semantic segmentation datasets.

Dataset	# images	resolution	# classes	size	Year
ISPRS Vaihingen ⁵	33	9 cm/px	6	$2,494 \times 2,064$ px	2013
ISPRS Potsdam ⁶	38	5 cm/px	6	$6,000 \times 6,000$ px	2013
Massachusetts [187]	1,322	100 cm/px	3	$1,500 \times 1,500$ px	2013
DFC15	7	5 cm/px	8	$10,000 \times 10,000$ px	2015
Zurich Summer [188]	20	62 cm/px	8	$1,000 \times 1,000$ px	2015
Inria Aerial [189]	360	30 cm/px	2	$5,000 \times 5,000$ px	2017
DLRS [190]	2,100	30 cm/px	17	256×256 px	2018
UAVid [191]	300	-	8	$3,968 \times 2,160$ px	2020
Hi-UCD [192]	1,293	10 cm/px	9	$1,024 \times 1,024$ px	2020
LandCover.ai [193]	41	25/50 cm/px	4	$6,307 \times 8,563$ px	2021

thematic map [186] and semantic segmentation datasets [141]. Table 3.2 summarizes publicly available multi-label aerial image datasets, and Figure 3.8 presents examples from DFC15-mul, BigEarthNet, and MLRSNet. UCM-mul and AID-mul are reproduced from UCM and AID, respectively, and their example images can be referred to in Figure 3.7.

In Table 3.3, we list several commonly-used and newly-published aerial image semantic segmentation datasets. As these datasets are produced in the laboratory circumstance, all pixels belonging to predefined classes are exhaustively annotated which is labor-consuming. Thus, I would like to thank all data producers for their great contributions to the community here. We can see that these images tend to have large sizes, and thus, cropping large-scale images into small patches with a sliding window is often taken as the first step of network training.

3.4.2 Evaluation metrics

To evaluate the network performance in single-scene recognition, Overall Accuracy (OA), Average Accuracy (AA), and Kappa coefficient are computed on test data with the following equations:

$$\begin{aligned}
 OA &= \frac{TP}{N}, \\
 AA &= \frac{1}{L} \sum_{c=1}^L \frac{TP_c}{N_c}, \\
 Kappa &= \frac{N \cdot TP - \sum_{c=1}^L (TP_c + FN_c) \cdot (TP_c + FP_c)}{N^2 - \sum_{c=1}^L (TP_c + FN_c) \cdot (TP_c + FP_c)},
 \end{aligned} \tag{3.3}$$

where TP represents the number of true positives counted over all test samples, of which the number is denoted as N . TP_c , FP_c , and FN_c indicate the number of true positives, false positives, and false negatives with respect to the c -th class, and N_c is the number of samples belonging to the c -th class. L is the number of scene classes. Among them, OA and Kappa assess a model from the perspective of its overall capacity, while AA is more sensitive to the class-wise performance. For instance, a model that performs poorly on rare classes can have a high OA but a low AA. Besides, OA can be equal to AA when

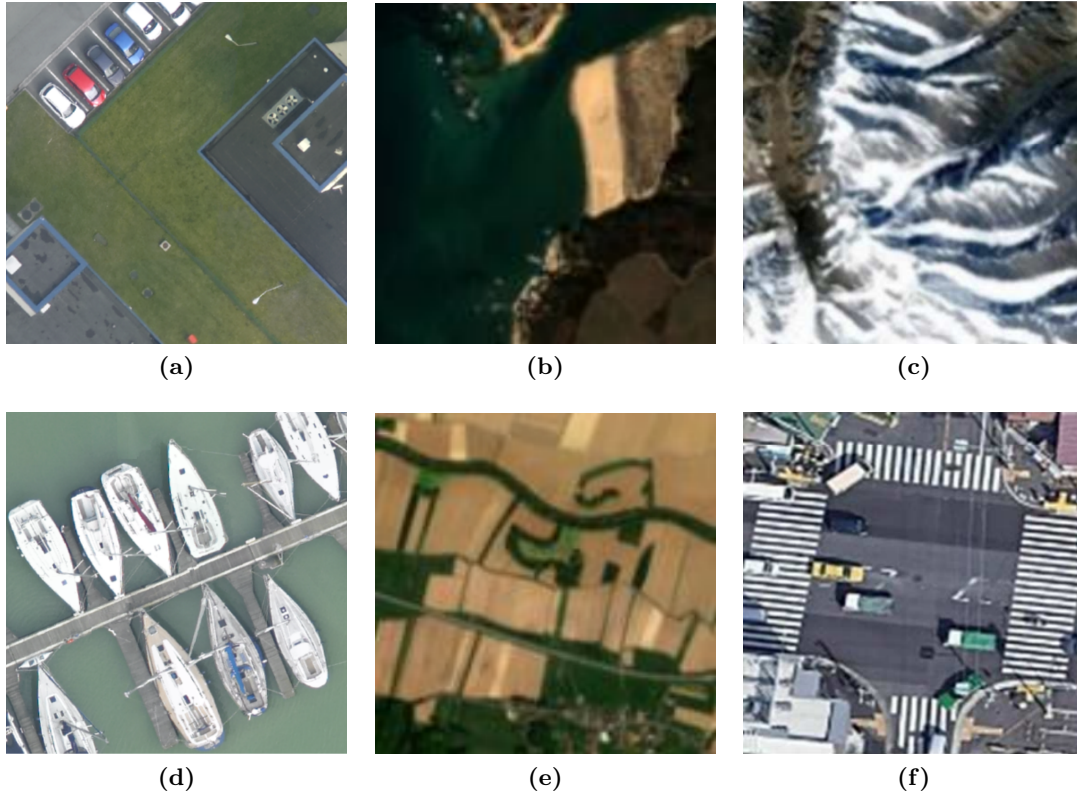


Figure 3.8: Example aerial images with multiple labels from DFC15-mul ((a) and (d)), BigEarth-Net ((b) and (c)), and MLRSNet ((e) and (f)) datasets. Their labels: (a) *Impervious, vegetation, building, and car*. (b) *permanently irrigated land, sclerophyllous vegetation, beaches, dunes, sands, estuaries, sea and ocean*. (c) *mountain, snow, and snowberg*. (d) *Water, clutter, and boat*. (e) *discontinuous urban fabric, non-irrigated arable land, land principally occupied by agriculture, and broad-leaved forest* (f) *buildings, crosswalk, grass, trees, cars, pavement, road, and intersection*.

the number of samples in each class is identical. Thus, it is more important to jointly compute AA and OA for a comprehensive evaluation on imbalanced datasets.

In contrast to single-scene recognition networks, multi-label object classification models are designed to make multiple predictions on each image. Hence, to evaluate such models, correctly inferring all labels of an image can not be simply counted as one correct prediction but should be counted multiple times depending on evaluation principles. To be more specific, there are three types of evaluation metrics for multi-label object classification networks:

- *Class-based Metrics:* Mean class-based precision (mCP), recall (mCR), F_1 (mCF₁) score, and per-class average precision (AP) are calculated for measuring the performance of networks from the perspective of class. Specifically, mCP, mCR, and mCF₁ score are computed as:

$$\begin{aligned} \text{mCP} &= \frac{1}{L} \sum_{c=1}^L \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c}, \\ \text{mCR} &= \frac{1}{L} \sum_{c=1}^L \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c}, \\ \text{mCF}_1 &= \frac{1}{L} \sum_{c=1}^L \frac{\text{TP}_c}{\text{TP}_c + \frac{1}{2}(\text{FP}_c + \text{FN}_c)}, \end{aligned} \quad (3.4)$$

where TP_c , FN_c , and FP_c represent numbers of true positives, false negatives, and false positives with respect to the c -th class, respectively. As to the per-class AP, we first rank all examples according to the predicted probability of the c -th class in each of them. Then we calculate the corresponding AP with the following formula:

$$\text{AP} = \frac{1}{N_c} \sum_{k=1}^N \frac{\text{TP}_c@k}{\text{TP}_c@k + \text{FP}_c@k} \times \text{rel}@k, \quad (3.5)$$

where N_c denotes the number of examples including the c -th class, and $\text{TP}_c@k$ and $\text{FP}_c@k$ represent numbers of true and false positives in top- k examples, respectively. Notably, $\text{TP}_c@k$ and $\text{FP}_c@k$ are equivalent to TP_c and FP_c , when k equals to N . $\text{rel}@k$ denotes the relevance between the k -th example and the c -th class, and it is set to 0/1 when the c -th class is included/excluded. Besides, the mean average precision (mAP) can be computed by averaging APs for all categories.

- *Example-based Metrics:* Mean example-based precision (mEP), recall (mER), and F_1 (mEF₁) score are computed to validate networks from the perspective of example with the following equations:

$$\begin{aligned} \text{mEP} &= \frac{1}{N} \sum_{k=1}^N \frac{\text{TP}_k}{\text{TP}_k + \text{FP}_k}, \\ \text{mER} &= \frac{1}{N} \sum_{k=1}^N \frac{\text{TP}_k}{\text{TP}_k + \text{FN}_k}, \\ \text{mEF}_1 &= \frac{1}{N} \sum_{k=1}^N \frac{\text{TP}_k}{\text{TP}_k + \frac{1}{2}(\text{FP}_k + \text{FN}_k)}, \end{aligned} \quad (3.6)$$

3 Aerial Scene Understanding in the Lab

where TP_k , FP_k , and FN_k denote numbers of true positives, false positives, and false negatives in the k -th example.

- *Overall Metrics:* Overall precision (OP), recall (OR), and F_1 (OF_1) score can be used to measure the performance of models from a more holistic perspective, and they are calculated as:

$$\begin{aligned} OP &= \frac{TP}{TP + FP}, \\ OR &= \frac{TP}{TP + FN}, \\ OF_1 &= \frac{TP}{TP + \frac{1}{2}(FP + FN)}, \end{aligned} \tag{3.7}$$

where TP, FP, and FN are counted based on predictions of all scenes and examples.

As to semantic segmentation networks, frequently used evaluation metrics are per-class CF_1 , mCF_1 , OA, and AA. Besides, Intersection over Union (IoU) is also computed with respect to each class for measuring overlapping areas between positive predictions and ground truths. The equation is as follows

$$IoU = \frac{TP_c}{TP_c + FP_c + FN_c}. \tag{3.8}$$

4 Aerial Scene Understanding in the Wild

In the field of aerial scene understanding, the mainstream research direction is to propose and validate algorithms under the laboratory prerequisites, where aerial images are well-cropped and fully annotated. To be more specific, in conventional scene recognition, aerial images are supposed to satisfy two requirements that target scenes are center-aligned and take the majority of respective images. Such constraints often result in images with monotonous scene-wise patterns and small scales. As to semantic segmentation of aerial imagery, dense pixel-level annotations are needed to convey sufficient supervisory signals for learning deep semantic segmentation networks. Since manually yielding pixel-level annotations is extremely arduous and expensive, there are limited datasets for semantic segmentation of aerial imagery, which restricts the applicability of existing studies in real-life applications. As a consequence, although many achievements have been attained in aerial scene understanding during recent years, the deployment of deep learning models in the wild is still a severe predicament. In order to take a further step towards the real-world scenario, we break these constraints in this dissertation by answering two questions: 1) What if images are collected freely and have large coverage? and 2) Can we deploy deep neural networks in practical applications at a low cost? As to the former, we propose a new task, namely multi-scene recognition, where images are collected without any constraints and algorithms should recognize all present scenes instead of only the dominant one. Regarding the latter, we focus on semantic segmentation networks, which are data-hungry and often suffer from insufficient annotations for novel tasks, and propose an annotation-friendly pipeline. The remainder of this chapter is organized as follows. Chapter 4.1 highlights the difference between single- and multi-scene recognition and briefly introduces current researches. Chapter 4.2 recalls the progress of semantic segmentation with incomplete labels, and eventually, a comprehensive view of existing datasets for these tasks is presented in Chapter 4.3.

4.1 Multi-scene Recognition

4.1.1 From Single- to Multi-scene Recognition

In Chapter 3.1, we have briefly introduced researches in the conventional scene classification task, where each aerial image is assigned only one scene label. During the literature review, it is not difficult to observe that most existing works share a common assumption that an aerial image contains only one scene, and thus evaluate the network performance on well-cropped single-scene aerial images. To produce single-scene image datasets, researchers usually resort to crowdsourcing platforms, e.g., OpenStreetMap (OSM), as they provide not only semantic attributes (e.g., category and function) but also geographical properties (e.g., geographic coordinate and geometrical shape) of ground targets. Fig 4.1 presents an example of querying features of a building on OSM. By simply setting coordi-

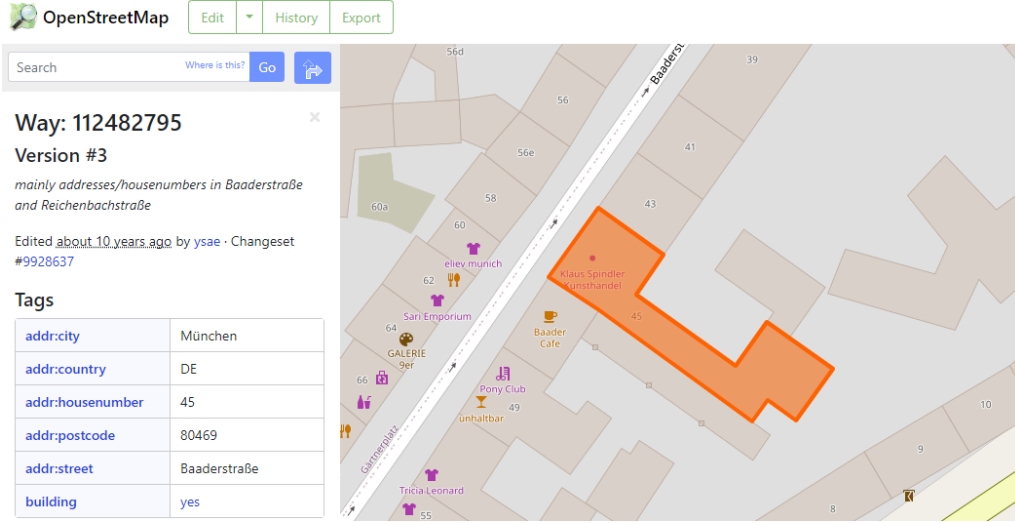


Figure 4.1: Illustration of querying building features on the OSM platform.

nates of the ground target, we can obtain the category, location, and shape of the queried building with a simple click. In [47], the authors delineate the process of generating single-scene image datasets. Specifically, thematic OSM layers with respect to scenes of interest are first extracted from the whole OSM database through semantic attribute filtering. By parsing the corresponding thematic layer, locations and shapes of entities belonging to each scene can be accessed and employed to determine where and what scale images should be captured. Once sufficient images are collected, their labels can be assigned automatically based on their semantic attributes. Figure 3.7 shows several examples of produced images and labels, and we can observe that in each aerial image, the target scene is located in the center and occupies most of the areas. However, in the real-world scenario, it is more often that there exist multiple scenes and distributed eccentrically in an aerial image. Hence, in this dissertation, we aim to study a more practical case, multi-scene recognition, aiming to identify all present scenes in unconstrained aerial images. In comparison with conventional scene recognition, this new task is different in terms of the following perspectives:

- *Objective.* In conventional aerial scene recognition, existing studies focus on identifying dominant scenes but neglect clutter scenes that are close to image borders or have negligible areas (see woodlands and residential at the fourth row of Figure 3.7). That is to say, even multiple scenes co-exist in one image, algorithms are designed to recognize only the dominant one. In contrast, the multi-scene recognition task aims to draw a comprehensive picture of present scenes and identify all but not one of them. Compared to single-scene recognition, this task is more challenging and able to exhaust the performance of deep neural networks in interpreting images of complex contents and sensing inconspicuous tiny targets.
- *Image acquisition.* As to single-scene recognition, all aerial images are well-cropped and have small coverage for ensuring the predominance of target scenes. In each aerial image, most of the objects are equipped with similar properties and correlated in a homogeneous pattern with respect to each scene. Besides, since target scenes are centered-aligned in major training samples, networks might overemphasize features

of objects in the image center. Nonetheless, in multi-scene recognition, images are captured in an unrestrained manner and allowed to cover large areas, which simulates the scenario in the wild that constrained images are not always available. Compared to single-scene images, unconstrained aerial images are prone to contain multiple co-occurring scenes, which are likely to be incomplete and trivial. Thus, multi-scene recognition is a more challenging and worthwhile research interest.

- *Label encoding.* In single-scene recognition, labels are encoded into one-hot vectors, representing probability distributions over all categories. In each one-hot vector, only the element corresponding to the target scene is 1 and the others are 0, which encourages algorithms to learn to pick the correct category from candidates with 100% confidence. As a consequence, the softmax function is often selected as the last activation function, as the sum of its outputs is 1, and each element indicates the probability of an image belonging to the corresponding scene. However, in multi-scene recognition, labels are multi-hot vectors, where each digit individually denotes the existence/presence of the corresponding scene. In this case, the sum of all elements is unforeseeable in advance, and the meaning of label vectors is different from that in single-scene recognition. In a multi-hot label vector, the value of each element suggests the probability distribution over the *presence* and *absence* but not all classes. As a consequence, instead of the softmax function, the sigmoid function is more frequently used as the last activation layer in deep networks.
- *Label semantics.* The encoding of multiple labels in multi-scene recognition and multi-label object classification is identical, but their semantics are far different. The concept of *scene* is of higher-level and more abstract in comparison with *object*. Besides, a scene is an association of multiple objects, and variant rearrangements of common objects can result in different scenes. Hence, images assigned common multiple object labels can have variant scene labels, which leads to the serious difficulty of thoroughly interpreting unconstrained aerial images.

4.1.2 Deep Learning for Multi-scene Recognition

As the first attempt [194], the authors propose a prototype-based memory network for constructing the prototype representation of each aerial scene and inferring multiple scene labels by measuring similarities between these prototypes and given multi-scene images. The insight of this work is that scenes appear similar structural, textural, and spectral patterns in high resolution aerial images even they are acquired by variant data platforms (see Figure 3.7). Therefore, an intuitive idea arises that deep neural networks can learn discriminative scene prototypes on single-scene aerial image datasets, which are abundant and readily accessed, in advance of inferring multiple co-occurring scenes in unconstrained aerial images. And this is expected to mitigate the problem of insufficient training data for multi-scene recognition. More details of this work can be referred to in Chapter 5.4. In [195], the authors treat multi-scene recognition as the multi-label problem and evaluate multi-label object classification networks on a benchmark dataset. Besides, single-scene classification networks are transferred to this task by substituting the sigmoid function for the softmax function, and their performance is validated as well. However, till now, multi-scene recognition remains underexplored.



Figure 4.2: Aerial images taken over urban residential areas in (a) Germany, (b) China, and (c) the US and are provided by Google Earth. Albeit identical functions, they share variant architectures, visual appearances, and layouts.

4.2 Semantic Segmentation of Aerial Imagery with Sparse Scribbled Annotations

4.2.1 From Dense to Sparse Pixel-wise Annotations

Semantic segmentation of aerial imagery refers to identifying the category of every pixel in high resolution aerial images and offers a pixel-level understanding of aerial images. With the great advancement of aerial photography techniques and deep learning-based methodologies, many achievements have been obtained in this field, and we have presented a brief review in Chapter 3.3. Albeit successful, these great successes are highly dependent on massive dense pixel-level annotations where all pixels are assigned their category through enormous manual visual inspections. Therefore, such annotations are expensive and at a substantial cost of time and human labor, which restricts the progress and deployment of existing researches in tasks suffering from data scarcity. Moreover, since deep learning-based approaches are data-driven and prone to overfit distributions of training data, the performance of trained models might show limited performance in interpreting test images that are collected from regions of variant cultural and natural environments. Figure 4.2 presents aerial images taken over Germany, China, and the United States (the US) with respect to the same scene, *residential*. It can be seen that construction styles and urban layouts are different, and it is not surprising that deep neural networks learned on one of the cultural zones may fail to interpret the others. As a consequence, in real-world applications, gathering aerial images of study areas and manually labeling them in a pixel-wise manner is a rule of thumb for putting deep learning-based models into practice. However, this pipeline suffers from the heavy annotation burden which hinders agile aerial scene understanding. To address this issue, we study a more annotation-friendly framework for semantic segmentation of aerial imagery based on incomplete and sparsely distributed labels. More specifically, instead of pixel-wise labeling aerial images, human annotators are required to label a few pixels by simply painting points, scribbles, or polygons within selected objects and assigning all pixels along or inside drawings uniform classes. Examples of sparse point-level, scribble(line)-level, and polygon-level annotations are shown in

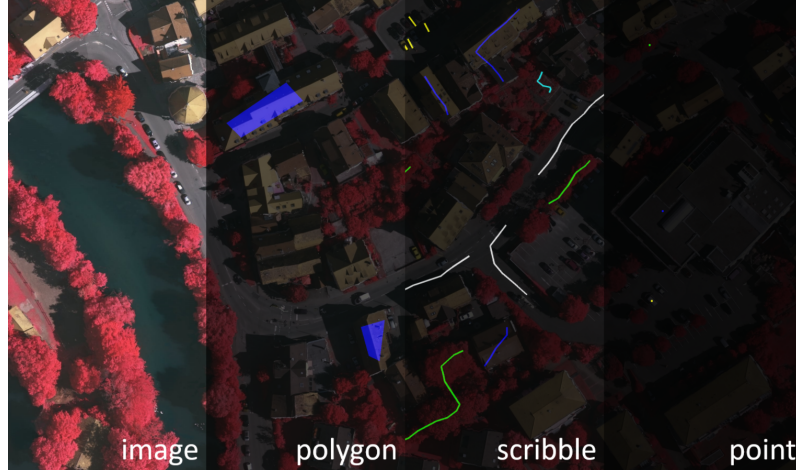


Figure 4.3: Illustration of sparse point-, scribble-, and polygon-level annotations.

Fig 4.3. Compared to dense pixel-wise labels, scribbled annotations are characterized by the following features:

- *Cheap acquisition.* Point-level annotations are yielded by drawing a dot per selected object, which mimics how humans refer to objects by pointing. Squiggles are easy to paint even for a child, and annotators only need to ensure that scribbled lines are painted inside objects so that pixels along one line can be assigned the same class label. Polygon-level annotations can be regarded as a special case of scribble-level annotations, where a line ends up at its starting point, and pixels located within one polygon are categorized into the same semantic class. In contrast to dense pixel-level labels, scribbled annotations are not required to fit complex object geometrical shapes, which significantly reduces the cost of time and human labor.
- *High confidence.* In the phase of yielding dense pixel-wise annotations, identifying pixels located at complex boundaries or in the shadow is not only time- and labor-consuming but also error-prone, especially when faced with natural objects. As shown in Figure 4.3, geometry shapes of trees are very complex, and pixels in the shadow are obscure and even arduous for remote sensing experts to correctly distinguish whether they belong to trees or roads. Nonetheless, in generating scribbled annotations, such difficulties can be avoided, as it is not mandatory to identify all pixels at boundaries or in the shadow. Therefore, annotators can just label explicit pixels which leads to the high confidence of scribbled labels and free of noisy supervisory signals.

4.2.2 Preliminaries

Albeit enjoying cheap acquisition and high confidence, scribbled annotations are sparse and disable fully supervised learning of semantic segmentation networks. Therefore, digging out semantics from image contexts plays a crucial role in learning with sparse scribbled annotations. To reach this goal, most of the existing researches [196, 197, 198] share a common assumption that pixels of homogeneous appearances (e.g., RGB values or intensities) and located nearby should contain identical semantics and be categorized into the

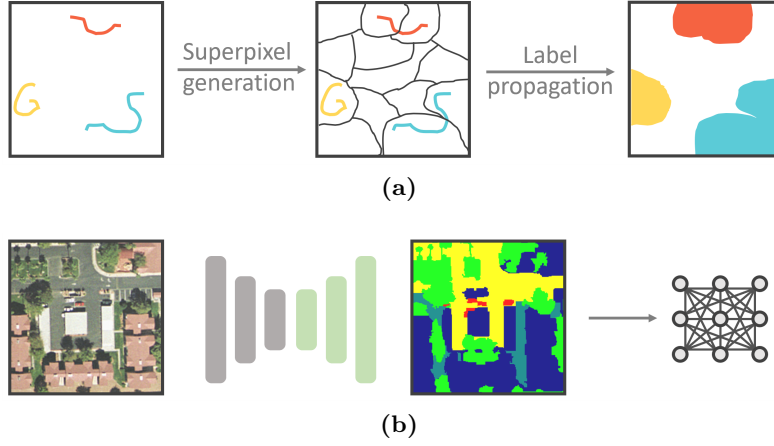


Figure 4.4: Illustration of two pipelines for learning with sparse segmentation. In (a), pixels are clustered into superpixels for label and semantic propagation before training networks. In (b), predicted masks are fed to a graph model for regularization based on input image contexts.

same class, and thus two pipelines are often followed: 1) clustering pixels into superpixels for label and semantic propagation and 2) regularizing predicted masks according to input image contexts. Figure 4.4 shows visual illustrations of the two pipelines. An example of the former is that the authors in [199] partition an image into superpixels and assign those overlapping scribbled annotations corresponding labels. Thus, semantic labels can be propagated from annotated pixels to unlabeled ones in common superpixels. As to the latter, the most frequent implementation is to regularize predictions with a graphical model, e.g., fully connected CRF [200], so that predicted segmentation masks are forced to accord with the spatial pattern of original images. Before diving into related literature, we briefly introduce Simple Linear Iterative Clustering (SLIC) [198] and fully connected CRF in this subsection.

SLIC is a simple yet computationally efficient superpixel generation algorithm. Specifically, an image is first projected into the CIELAB color space¹, and each pixel is represented as a vector consisting of lightness (L), red/green value (a), blue/yellow value (b), and coordinates (x, y). Afterwards, superpixels are generated by clustering pixel vectors through K-Means [201] but within a limited search region, and output clusters are so-called superpixels. Initially, k cluster centers are evenly sampled from an image, and the size of the search region of each center is defined as four times larger than the superpixel size. Pixels are assigned to the nearest cluster centers within their search regions. Once all pixels are sorted out, cluster centers are updated by calculating the mean of pixel vectors located inside. These processes should be iteratively conducted until the distance between the new and previous center is smaller than a certain threshold.

Fully connected CRF, also known as dense CRF, maximizes label agreement between adjacent and similar-looking pixels by minimizing their Gibbs energy. The energy function is defined as:

$$E = \sum_i \theta_u(x_i) + \sum_{ij} \theta_p(x_i, x_j), \quad (4.1)$$

¹International Commission on Illumination (Commission Internationale de l’Eclairage) (CIE). <http://cie.co.at/>

4.2 Semantic Segmentation of Aerial Imagery with Sparse Scribbled Annotations

where $\theta_u(x_i)$ is the unary potential and calculated as $\theta_u(x_i) = -\log P(x_i)$. Here x_i ranges over all pixels of an image, and $P(x_i)$ indicates the label probability of pixel i . $\theta_p(x_i, x_j)$ measures pairwise potentials between pixel i and j and in our case, we only adopt two Gaussian kernels. Thus, the pairwise potential can be computed with the following equation:

$$\theta_p(x_i, x_j) = \mu(x_i, x_j)(w_1 k_1 + w_2 k_2), \quad (4.2)$$

and k_1 and k_2 are calculated as:

$$\begin{aligned} k_1 &= \exp\left(-\frac{\|p_i - p_j\|^2}{2\theta_1^2} - \frac{\|I_i - I_j\|^2}{2\theta_2^2}\right), \\ k_2 &= \exp\left(-\frac{\|p_i - p_j\|^2}{2\theta_3^2}\right), \end{aligned} \quad (4.3)$$

where p_i and I_i indicate the position and color intensity of pixel i . θ_1 , θ_2 , and θ_3 are hyperparameters that control the kernel “scale”. In Eq. 4.3, k_1 is known as *appearance kernel* and tends to classify adjacent pixels with comparable appearances, i.e., color intensities, into the same classes, while k_2 , so-called *smoothness kernel*, penalizes pixels nearby but assigned different labels. In Eq. 4.2, $\mu(x_i, x_j)$ is a label compatibility term and penalizes pairs of pixels nearby but categorized into contradictory classes, such as **car** and **river** or **desert** and **ship**. As a consequence, predicted masks become smoother within homogeneous areas and more semantically compatible in local regions.

4.2.3 Learning with Sparse Scribbled Annotations

Sparse scribbled annotations are easy to obtain and of high confidence. Nonetheless, there are very limited researches in semantic segmentation with such annotations, and among them, semi-supervised learning is dominant. In [202], point-level annotations are first employed to train semantic segmentation networks, e.g., FCN, on natural images with auxiliary abjectness [203], which indicates how likely each pixel belongs to foreground objects or the background. In [199], the authors make the first attempt to learn semantic segmentation networks under scribble-supervision and propagate semantic labels from squiggles to unlabeled pixels through a graphical model built on superpixels of training images. Inspired by successes in segmenting natural images, research interests in applying scribbled annotations to semantic segmentation of aerial imagery are arising, and several efforts [196, 197] have been deployed in recent years. Wu et al. [196] study the effectiveness of scribble-level annotations in building footprint segmentation and synthesize obb-scribble masks by fitting oriented bounding boxes around scribbles. Afterwards, an adversarial architecture is employed to generate building footprint masks from aerial images and enforce them to resemble obb-scribble masks. In [197], the authors aim to tackle the problem of inaccurate boundary prediction that results from training networks on polygon-level annotations and employ a fully connected CRF to refine predicted segmentation masks. To address the same issue, Lu et al. [204] generate pseudo annotations around object boundaries by replacing regions along predicted boundaries with contents randomly cropped from other predicted masks, and both pseudo and scribbled annotations are used to learn semantic segmentation networks. In [205], point- and scribble-level annotations are treated as low-cost supplementary data and taken as input for generating relevance maps, where high values indicate strong relevance between unlabeled pixels and scribbled annotations. Afterwards, relevance maps, aerial images, and feature maps extracted by

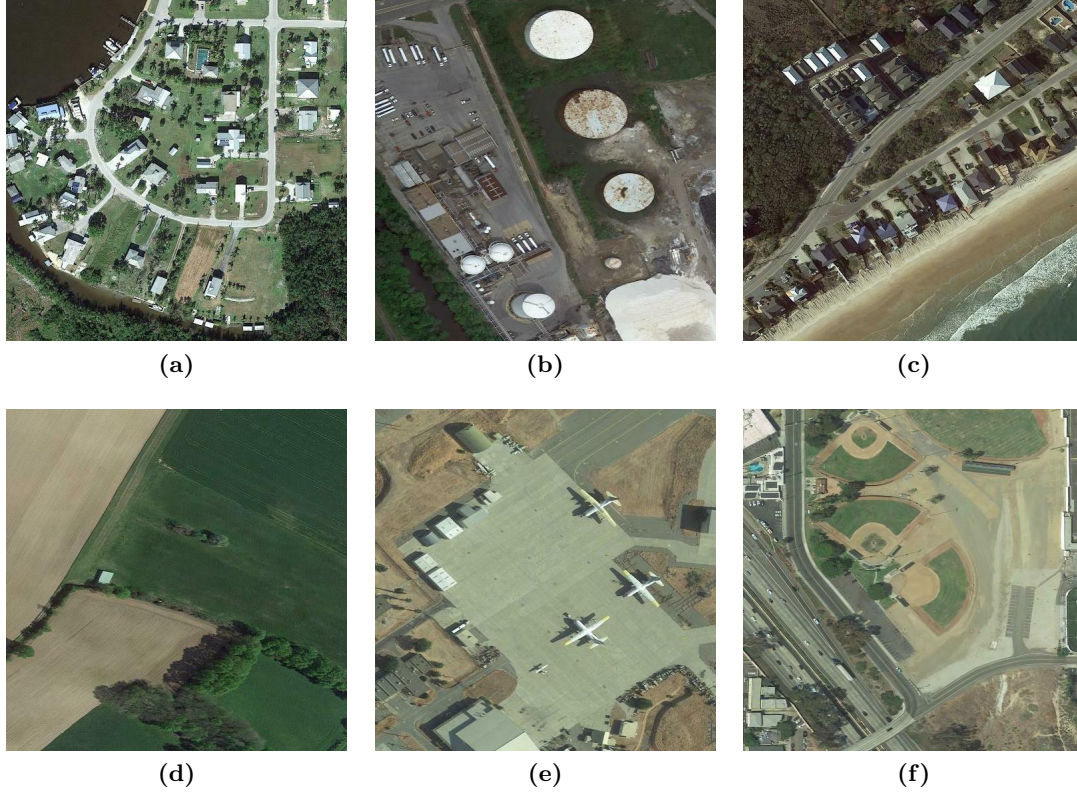


Figure 4.5: Example unconstrained aerial images in the MAI dataset. Their scene-level multiple labels are here: (a) farmland and residential; (b) baseball, woodland, parking lot, and tennis court; (c) commercial, parking lot, and residential; (d) woodland, residential, river, and runway; (e) river and storage tanks; (f) beach, woodland, residential, and sea.

a pre-trained VGG are fed to the proposed attention-guided multi-scale segmentation network for inferring corresponding masks. In [206], the authors learn to extract road surfaces under the supervision of road centerline-like line-level annotations. Specifically, SLIC is employed to generate superpixels of input images, and then a graph is built on superpixels to produce road proposal masks and broadened road centerlines. Afterwards, the proposed scribble-based weakly supervised road surface extraction method, namely ScRoadExtractor, is trained to simultaneously predict road proposal masks and boundary masks, which are extracted by Holistically-nested Edge Detection (HED) algorithm [207]. Compared to the progress of semantic segmentation with dense pixel-wise annotations, researches about incorporating sparse and incomplete labels are far from sufficient, and this task showcases great potential in real-world applications.

4.3 Data and Evaluation Metrics

In contrast to the booming development of single-scene aerial image datasets, existing multi-scene aerial image datasets are extremely scarce. In [194], the authors propose a multi-scene dataset, namely MAI, where 3923 aerial images are taken from Google Earth imagery over Europe and North America. The size of each image is 512×512 , and spatial

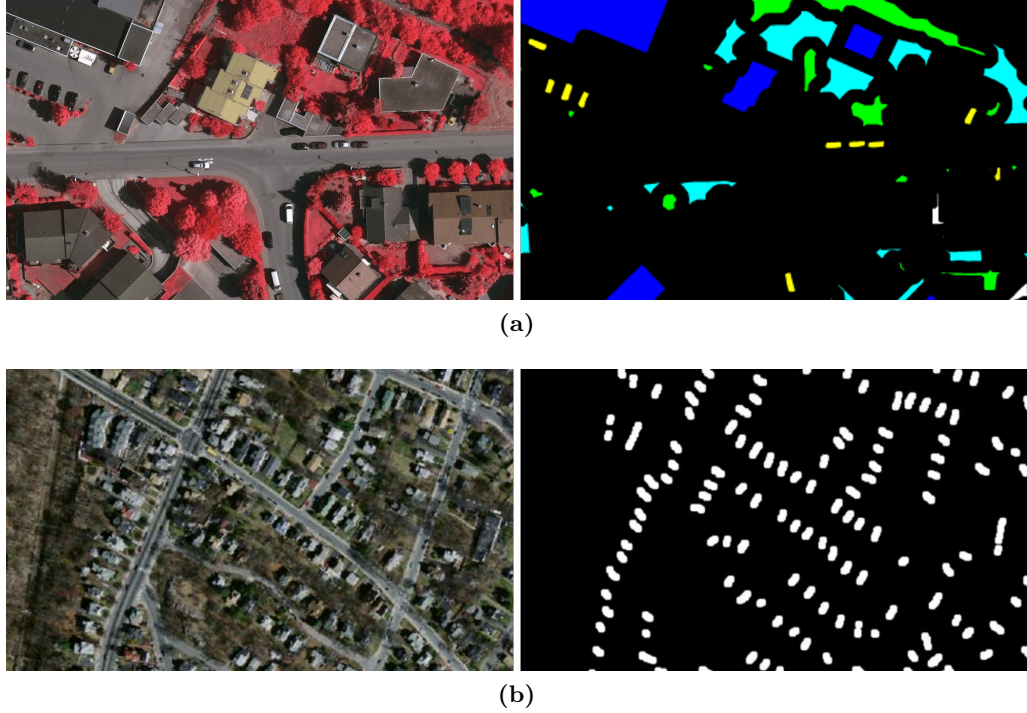


Figure 4.6: Example scribbled pixel-wise annotations. (a) is reproduced from the Massachusetts Buildings dataset [208]. (b) is generated by discarding labels in the ISPRS Vaihingen dataset². Black pixels are unlabeled, and those with other colors are assigned semantic labels.

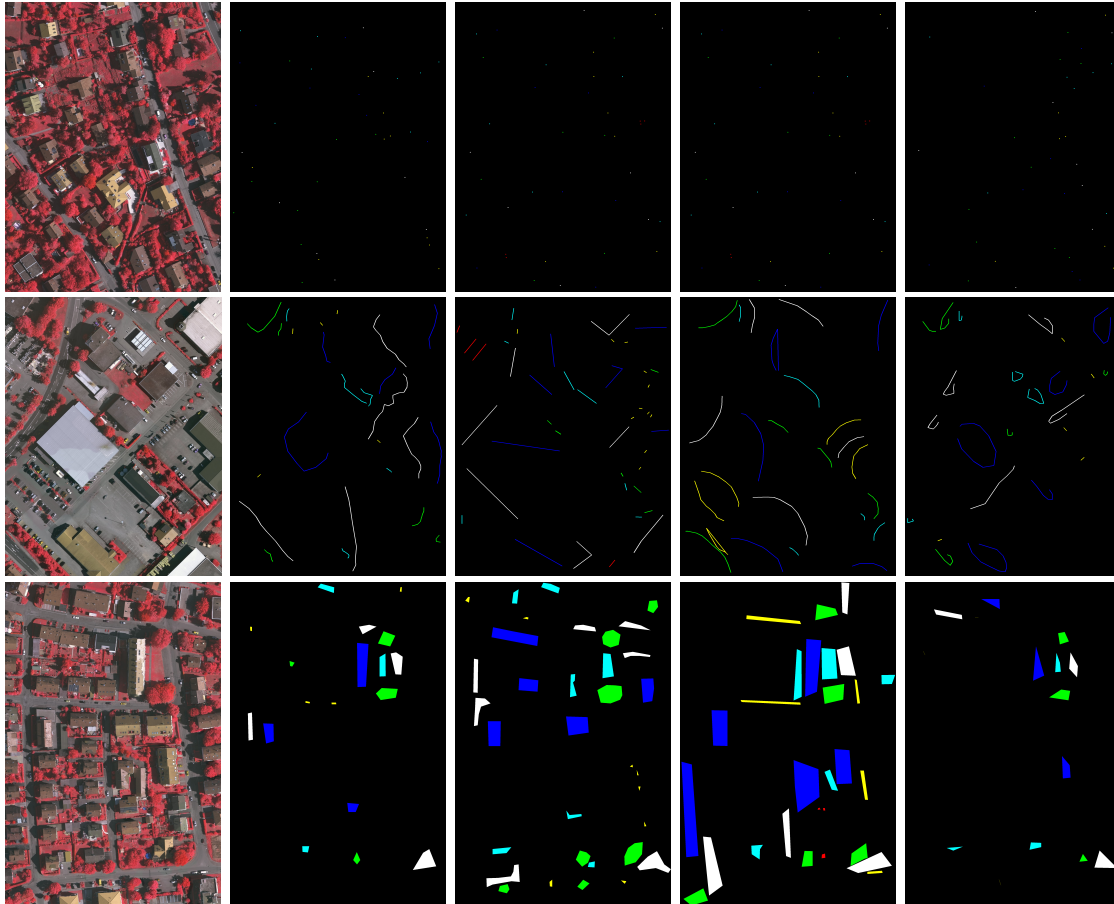
resolutions vary from 0.3 m/pixel to 0.6 m/pixel. The authors define 24 scene labels, including apron, baseball, beach, commercial, farmland, woodland, parking lot, port, residential, river, storage tanks, sea, bridge, lake, park, roundabout, soccer field, stadium, train station, works, golf course, runway, sparse shrub, and tennis court. Each aerial image is then visually inspected and assigned multiple scene labels according to their contents. Figure 4.5 shows several samples. Following this work, a large-scale multi-scene aerial image dataset, termed MultiScene, is proposed in [195]. In total, 100,000 unconstrained aerial images are cropped from Google Earth imagery and cover Europe, Asia, North America, South America, Africa, and Oceania. The number of scene classes is increased to 36, and 14,000 images are manually annotated. Besides, all images are assigned with labels crawled from OSM. Since the MultiScene dataset is the major contribution of our fourth work, we present a more specific introduction in Chapter 5.4. Since multi-scene recognition is, in essence, a multi-label problem, and thus, the evaluation metrics introduced in Chapter 3.4 are also applied to validate the performance of multi-scene recognition networks.

For evaluating the performance of networks learned on incomplete scribbled labels, current researches mainly generate synthetic labels by conducting morphological transformations, i.e., erosion, on dense pixel-wise annotations provided in existing semantic segmentation datasets. For example, Maggiolo et al. [197] morphologically discard 60% of original labels, especially those of boundary pixels (cf. Figure 4.6a). Besides, Wu et al. [196] experiment with both automatic and manual scribble generation, but only one semantic class is taken into consideration (see Figure 4.6b). Recently, in [194], the authors ask four human annotators (two remote sensing experts and two non-experts)

Table 4.1: The total numbers of pixels labeled with sparse point-, line-, and polygon-level annotations (middle three columns) and dense annotations (right column) in the Vaihingen and Zurich Summer datasets.

Dataset Name	Point	Line	Polygon	Dense*
ISPRS Vaihingen	18,787	480,593	4,591,409	54,373,518
Zurich Summer	29,508	330,767	1,445,270	12,266,287

*Background/Clutter is not counted.

**Figure 4.7:** Example scribbled pixel-wise annotations of the ISPRS Vaihingen dataset. The 2nd and 3rd columns are made by experts, and the left two are created by non-experts. Legend—white: impervious surfaces, blue: buildings, cyan: low vegetation, green: trees, yellow: cars.

to relabel the ISPRS Vaihingen ³ and Zurich summer [209] datasets with points, scribbles, and polygons. Instruction is given before annotation to ensure that non-experts are equipped with primary knowledge of data labeling. Figure 4.7 shows example annotations created by four annotators, and Table 4.1 compares the number of yielded incomplete annotations and original dense labels. To evaluate the performance of network trained on scribbled annotates, the evaluation metrics mentioned in Chapter 3.4 can be leveraged

³<https://www2.isprs.org/commissions/comm2/wg4/benchmark/semantic-labeling/>

as well. Besides, it is interesting to see that for scenes sharing homogeneous appearance, networks trained on incomplete labels can achieve comparable performance in comparison with those trained on dense labels (see the column *tree* in Table 5.5).

5 Summary of works

To reach the objectives of this thesis, the author deploy efforts in the following four aspects: CNN

- A hybrid convolutional and bidirectional LSTM network is proposed for categorizing multiple objects present in an aerial scene image. The network is featured by modeling label dependencies through a bi-directional LSTM, and details are introduced in Chapter 5.1.
- Relation networks are introduced to the multi-label object classification task, and an attention-aware label relational reasoning network is proposed and detailed in Chapter 5.2.
- To facilitate the progress of multi-label object classification, two high resolution aerial image datasets, namely the DFC15 and AID multilabel datasets, are created and made publicly available. Chapter 5.1 and 5.2 describe these two datasets, respectively.
- A prototype-based memory network is proposed for multi-scene recognition, and it mimics how humans perceive complex scenes by learning and memorizing individual scenes in advance. Chapter 5.3 delineates the composition and mechanism of the proposed network.
- Two multi-scene aerial image datasets, termed as MAI and MultiScene datasets, are published and introduced in Chapter 5.3 and 5.4, respectively. In the latter dataset, not only manual annotations but also crowdsourced annotations are provided which enables researches in network learning from noisy labels for multi-scene recognition.
- In Chapter 5.5, a framework for semantic segmentation of aerial images based on incomplete annotations is described. In contrast to previous studies where synthetic scribbled labels are used, this work evaluates the effectiveness of the pipeline in a real-world scenario where four annotators (including two non-experts) are asked to manually yield point-, line-, and polygon-level annotations.

In this chapter, a brief summary of the five peer-reviewed articles made by the author (as the first author) is presented.

5.1 Exploiting label correlations with bidirectional LSTM for multi-label object classification

5.1.1 Motivation

Current researches [210, 126, 211, 131] in aerial image multi-label object classification deploy limited efforts to model inherent correlations between various objects for identifying

co-existing objects. However, in the real-life world, certain object categories can have strong relevances, for example, cars are often driven or parked on pavements, and ships are piloted or harbored on the water in most cases. To demonstrate the class dependency, we calculate conditional probabilities for each of two categories. Let C_r denote referenced class, and C_p denote potential co-occurrence class. Conditional probability $P(C_p|C_r)$, which depicts the possibility that C_p exhibits in an image, where the existence of C_r is priorly known, can be solved with Eq. 5.1,

$$P(C_p|C_r) = \frac{P(C_p, C_r)}{P(C_r)}. \quad (5.1)$$

$P(C_p, C_r)$ indicates the joint occurrence probability of C_p and C_r , and $P(C_r)$ refers to the priori probability of C_r . Conditional probabilities of all class pairs in UCM multi-label datasets are listed in Figure 5.1, and it is intuitive that some classes have strong dependencies. For instance, it is highly possible that there are pavements in images, which contain airplanes, buildings, cars, or tanks. Moreover, it is notable that class dependencies are not symmetric due to their particular properties. For example, $P(sea|ship)$ is twice as $P(ship|sea)$ due to the reason that the occurrence of ships always infer to the co-occurrence of sea, while not vice versa. Therefore, to thoroughly dig out the correlation among various classes, it is crucial to model class probabilistic dependencies bidirectionally in a classification method.

To this end, we boil the multi-label classification down into a structured output problem, instead of a simple regression issue, and employ a unified framework of a CNN and a bidirectional Recurrent Neural Network (RNN) to 1) extract semantic features from raw images and 2) model image-label relations as well as bidirectional class dependencies, respectively.

5.1.2 Methodology

The proposed CA-Conv-BiLSTM, as illustrated in Figure 5.2, is composed of three components: a feature extraction module, a class attention learning layer, and a Bidirectional LSTM-based recurrent sub-network. More specifically, the feature extraction module employs a stack of interleaved convolutional and pooling layers to extract high-level features, which are then fed into a class attention learning layer to produce discriminative class-specific features. Afterwards, a bidirectional LSTM-based recurrent sub-network is attached to model both probabilistic class dependencies and underlying relationships between image features and labels.

Dense High-level Feature Extraction. Learning efficient feature representations of input images is extremely crucial for image classification task. To this end, a modern popular trend is to employ a CNN architecture to automatically extract discriminative features, and thus, our model adapts VGG-16 [10], one of the most welcoming CNN architectures for its effectiveness and elegance, to extract high-level features for our task.

Specifically, the feature extraction module consists of 5 convolutional blocks, and each of them contains 2 or 3 convolutional layers (as illustrated in the left of Figure 5.2). Notably, the number of filters is equivalent in a common convolutional block and doubles after the spatial dimension of feature maps is scaled down by pooling layers. The receptive field of all convolutional filters is 3×3 , which increases nonlinearities inside the feature extraction module. Besides, the convolution stride is 1 pixel, and the spatial padding of each convolutional layer is set as 1 pixel as well. Among these convolutional blocks,

5.1 Exploiting label correlations with bidirectional LSTM for multi-label object classification

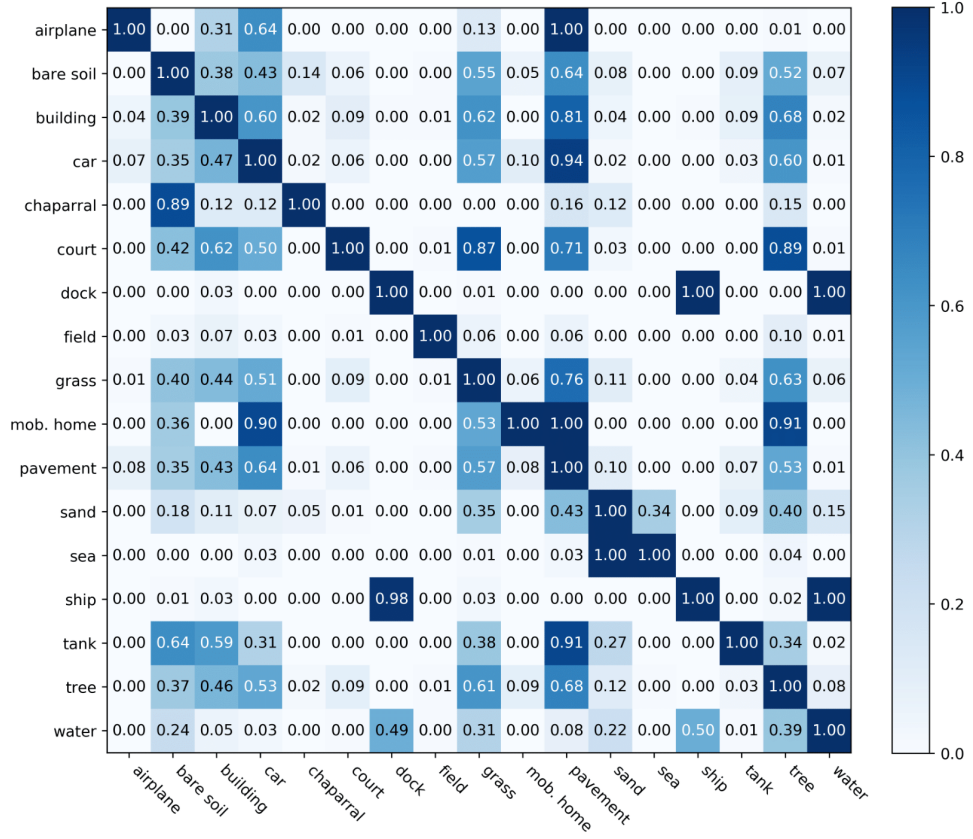


Figure 5.1: The contribution matrix of labels in UCM dataset. Labels at X-axis represent referenced classes C_r , while labels at Y-axis are potential co-occurrence classes C_p . Conditional probabilities $P(C_p|C_r)$ of each class pair are present in corresponding blocks.

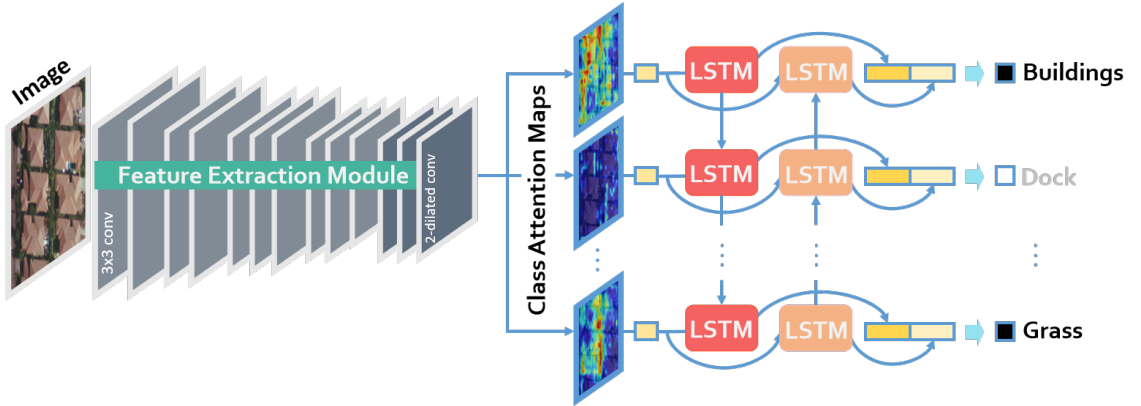


Figure 5.2: The architecture of the proposed CA-Conv-BiLSTM for the multi-label classification of aerial images.

max-pooling layers are interleaved for reducing the size of feature maps and meanwhile, maintaining only local representative, such as, maximum in a 2×2 -pixel region. The size of pooling windows is 2×2 pixels, and the pooling stride is 2 pixels, which halves feature maps in width and length.

Features directly learned from a conventional CNN (e.g., VGG-16) are proved to be high-level and semantic, but their spatial resolution is significantly reduced, which is not favorable for generating high-dimensional class-specific features in the subsequent class attention learning layer. To address this, max-pooling layers following the last two convolutional blocks are discarded in our model, and atrous convolutional filters with dilation rate 2 are employed in the last convolutional block for preserving original receptive fields. Consequently, our feature extraction module is capable of learning high-level features with finer spatial resolution, so called “dense”, compared to VGG-16, and it is feasible to initialize our model with pre-trained VGG-16, considering that all filters have equivalent receptive fields.

Moreover, it is worth nothing that other popular CNN architectures can be taken as prototypes of the feature extraction module, and thus, we extend researches to GoogLeNet [12] and ResNet [15] for a comprehensive evaluation of CA-Conv-BiLSTM. Regarding GoogLeNet, i.e., Inception-v3 [11], the stride of convolutional and pooling layers after “mixed7” is reduced to 1 pixel, and the dilation rate of convolutional filters in “mixed9” is 2. For ResNet (we use ResNet-50), the convolution stride in last two residual blocks is set as 1 pixel, and the dilation rate of filters in the last residual block is 2. Besides, layers after global average pooling layers, as well as itself, are removed to ensure dense high-level feature maps.

Class Attention Learning Layer. Although Features extracted from pre-trained CNNs are high-level and can be directly fed into a fully connected layer for generating multi-label predictions, it is infeasible to learn high-order probabilistic dependencies by recurrently feeding it with identical features. Therefore, extracting discriminative class-wise features plays a key role in discovering class dependencies and effectively bridging CNN and RNN for multi-label classification tasks.

Here, we propose a class attention learning layer to explore features with respect to each category, and the proposed layer, illustrated in the middle of Figure 5.2, consists of the following two stages: 1) generating class attention maps via a 1×1 convolutional layer with stride 1, and 2) vectorizing each class attention map to obtain class-specific features. Formally, given feature maps \mathbf{X} , extracted from the feature extraction module, with a size of $W \times W \times K$, and let \mathbf{w}_l represent the l -th convolutional filter in the class attention learning layer. The attention map \mathbf{M}_l for class l can be obtained with the following formula:

$$\mathbf{M}_l = \mathbf{X} * \mathbf{w}_l, \quad (5.2)$$

where l ranges from 1 to the number of classes. Besides, $*$ represents convolution operation. Given that the size of convolutional filters is 1×1 , and the stride is 1, Eq. 5.2 can be further modified as:

$$\mathbf{M}_l(p, q) = \sum_{k=1}^K w_{l,k} \mathbf{X}_k(p, q), \quad (5.3)$$

where $p, q = 1, 2, \dots, W$, and $\mathbf{M}_l(p, q)$ and $\mathbf{X}_k(p, q)$ indicate activations of the class attention map \mathbf{M}_l and the k -th channel of \mathbf{X} at a spatial location (p, q) , respectively. $w_{l,k}$ is the k -th channel of \mathbf{w}_l . The modified formula highlights that a class attention map \mathbf{M}_l is intrinsically a linear combination of all channels in \mathbf{X} , and $w_{l,k}$ depicts the importance of the k -th channel of \mathbf{X} for class l . Therefore, $\mathbf{M}_l(p, q)$ with a strong activation suggests that the region is highly relevant to class l , and vice versa.

Subsequently, class attention maps \mathbf{M}_l are transformed into class-wise feature vectors \mathbf{v}_l of W^2 dimensions by vectorization. Instead of fully connecting class attention maps to each hidden unit in the following layer, we construct class-wise connections between class attention maps and their corresponding hidden units, i.e., corresponding time steps in a LSTM layer in our network. In this way, features fed into different units are retained to be class-specific discriminative and significantly contribute to exploitation of the dynamic class dependency in the subsequent bidirectional LSTM layer.

Class Dependency Learning via a BiLSTM-based Sub-network. As an important branch of neural networks, RNN is widely used in dealing with sequential data, e.g., textual data and temporal series, due to its strong capabilities of exploiting implicit dependencies among inputs. Unlike CNN, RNN is characterized by its recurrent neurons, of which activations are dependent on both current inputs and previous hidden states. However, conventional RNNs suffer from the gradient vanishing problem and are found difficult to learn long-term dependencies. Therefore, in this work, we seek to model class dependencies with an LSTM-based RNN.

Instead of directly summing up inputs as in a conventional recurrent layer, an LSTM layer relies on specifically designed hidden units, LSTM units, where information, such as the class dependency between category l and $l - 1$, is “memorized”, updated, and transmitted with a memory cell and several gates. Specifically, given a class-specific feature \mathbf{v}_l obtained from the class attention learning layer as an input of the LSTM memory cell \mathbf{c}_l at time step l , and let \mathbf{h}_l represent the activation of \mathbf{c}_l . New memory information $\tilde{\mathbf{c}}_l$, learned from the previous activation \mathbf{h}_{l-1} and the present input feature \mathbf{v}_l , is obtained as follows:

$$\tilde{\mathbf{c}}_l = \tanh(\mathbf{W}_{cv}\mathbf{v}_l + \mathbf{W}_{ch}\mathbf{h}_{l-1} + \mathbf{b}_c), \quad (5.4)$$

where \mathbf{W}_{cv} and \mathbf{W}_{ch} denote weight matrix from input vectors to memory cell and hidden-memory coefficient matrix, respectively, and \mathbf{b}_c is a bias term. Besides, $\tanh(\cdot)$ is the hyperbolic tangent function. In contrast to conventional recurrent units, where the $\tilde{\mathbf{c}}_l$ is directly used to update the current state \mathbf{h}_l , an LSTM unit employs an input gate \mathbf{i}_l to control the extent to which $\tilde{\mathbf{c}}_l$ is added, and meanwhile, partially omits uncorrelated prior information from \mathbf{c}_{l-1} with a forget gate \mathbf{f}_l . The two gates are performed by the following equations:

$$\begin{aligned} \mathbf{i}_l &= \sigma(\mathbf{W}_{iv}\mathbf{v}_l + \mathbf{W}_{ih}\mathbf{h}_{l-1} + \mathbf{W}_{ic}\mathbf{c}_{l-1} + \mathbf{b}_i), \\ \mathbf{f}_l &= \sigma(\mathbf{W}_{fv}\mathbf{v}_l + \mathbf{W}_{fh}\mathbf{h}_{l-1} + \mathbf{W}_{fc}\mathbf{c}_{l-1} + \mathbf{b}_f). \end{aligned} \quad (5.5)$$

Consequently, the memory cell \mathbf{c}_l is updated by

$$\mathbf{c}_l = \mathbf{i}_l \odot \tilde{\mathbf{c}}_l + \mathbf{f}_l \odot \mathbf{c}_{l-1}, \quad (5.6)$$

where \odot represents element-wise multiplication. Afterwards, an output gate \mathbf{o}_l , formulated by

$$\mathbf{o}_l = \sigma(\mathbf{W}_{ov}\mathbf{v}_l + \mathbf{W}_{oh}\mathbf{h}_{l-1} + \mathbf{W}_{oc}\mathbf{c}_l + \mathbf{b}_o), \quad (5.7)$$

is designed to determine the proportion of memory content to be exposed, and eventually, the memory cell \mathbf{c}_l at time step l is activated by

$$\mathbf{h}_l = \mathbf{o}_l \tanh(\mathbf{c}_l). \quad (5.8)$$

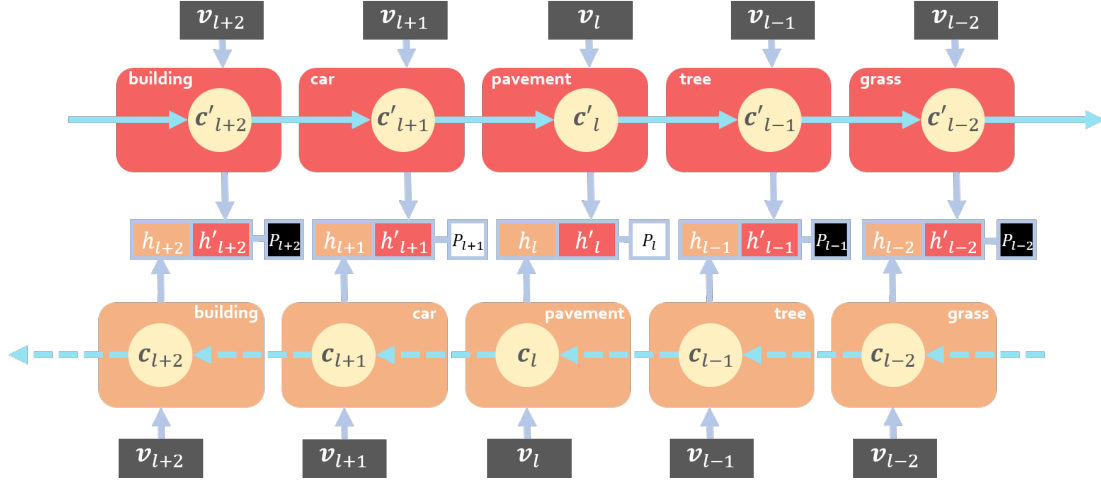


Figure 5.3: Illustration of the bidirectional structure. The direction of the upper stream is opposite to that of the lower stream. Notably, h'_{l-1}, c'_{l-1} denotes the activation and memory cell in the upper stream at the time step, which corresponds to class $l-1$ for convenience (considering that the subsequent time step is usually denoted as $l+1$).

Although it is not difficult to discover that the activation of the memory cell at each time step is dependent on both input class-specific feature vectors and previous cell states. However, taking into account that the class dependency is bidirectional, as demonstrated in Section 5.1.1, a single-directional LSTM-based RNN is insufficient to draw a comprehensive picture of inter-class relevance. Therefore, a bidirectional LSTM-based RNN, composed of two identical recurrent streams but with reversed directions, is introduced in our model, and the hidden units are updated based on signals from not only their preceding states but also subsequent ones.

In order to practically adapt a bidirectional LSTM-based RNN to modeling the class dependency, we set the number of time steps in our bidirectional LSTM-based sub-network equivalent to that of classes under the assumption that distinct classes are predicted at respective time steps. Such design enjoys two outstanding characteristics: on one hand, the LSTM memory cell at time step l , c_l , focuses on learning dependent relationship between class l and others in dual directions (cf. Figure 5.3), and on the other hand, the occurrence probability of class l , P_l , can be predicted from outputs $[h_l, h'_l]$ with a single-unit fully connected layer:

$$P_l = \sigma(\mathbf{w}_l[h_l, h'_l] + \mathbf{b}_l), \quad (5.9)$$

where h'_l denotes the activation of c_l in the other direction, and σ is used as the activation function.

5.1.3 Results

Table 5.1 exhibits results on UCM multi-label dataset, and it can be seen that compared to directly applying standard CNNs to multi-label classification, CA-Conv-LSTM framework performs superiorly as expected due to taking class dependencies into consideration. Mostly enjoying this framework, CA-GoogLeNet-LSTM achieves the best mean F_1 score of 81.78% and an increment of 1.10% in comparison with other CA-Conv-LSTM models

Table 5.1: Quantitative Results on UCM Multi-label Dataset (%)

Model	mEF ₁	mEF ₂	mEP	mER	mCP	mCR
VGGNet [10]	78.54	80.17	79.06	82.30	86.02	80.21
VGG-RBFNN [126]	78.80	81.14	78.18	83.91	81.90	82.63
CA-VGG-LSTM	79.57	80.75	80.64	82.47	87.74	75.95
CA-VGG-BiLSTM	79.78	81.69	79.33	83.99	85.28	76.52
GoogLeNet [12]	80.68	82.32	80.51	84.27	87.51	80.85
GoogLeNet-RBFNN [126]	81.54	84.05	79.95	86.75	86.19	84.92
CA-GoogLeNet-LSTM	81.78	85.16	78.52	88.60	86.66	85.99
CA-GoogLeNet-BiLSTM	81.82	84.41	79.91	87.06	86.29	84.38
ResNet-50 [15]	79.68	80.58	80.86	81.95	88.78	78.98
ResNet-RBFNN [126]	80.58	82.47	79.92	84.59	86.21	83.72
CA-ResNet-LSTM	81.36	83.66	79.90	86.14	86.99	82.24
CA-ResNet-BiLSTM	81.47	85.27	77.94	89.02	86.12	84.26

mEF₂ indicate the mean example-based F_2 score.

and GoogLeNet, respectively. Concerning the signification of employing a bidirectional structure, CA-Conv-BiLSTM performs better than CA-Conv-LSTM in the mean F_1 score, and compared to Conv-RBFNN, our models achieve higher mean F_1 and F_2 scores, increased by at most 0.98% and 2.80%, respectively. Another important observation is that our proposed model is equipped with higher example-based recall but lower example-based precision, which leads to a relatively higher mean F_2 score.

In addition to validate classification capabilities of the network by computing the mean F_2 score, we further explore the effectiveness of class-specific features learned from the proposed class attention learning layer and try to “open” the black box of our network by feature visualization. Example class attention maps produced by the proposed network on UCM multi-label dataset are shown in Figure 5.4. As we can see, these maps highlight discriminative regions for positive classes, while present almost no activations when corresponding objects are absent in original images. For example, object labels of the image at the first row in Figure 5.4 are building, grass, pavement, and tree, and its class attention maps for these categories are strongly activated.

We also evaluate our network on the DFC15 multi-label dataset. The dataset is constructed based on a semantic segmentation dataset, DFC15 (see Chap 3.4.1). we crop its tiles into images of 600×600 pixels with a 200-pixel-stride sliding window and discard images containing unclassified pixels. Labels of each image are yielded by aggregating its included pixel-level labels. For more experimental results and technical details on DFC15 as well as UCM multi-label datasets, please refer to **Appendix A**.

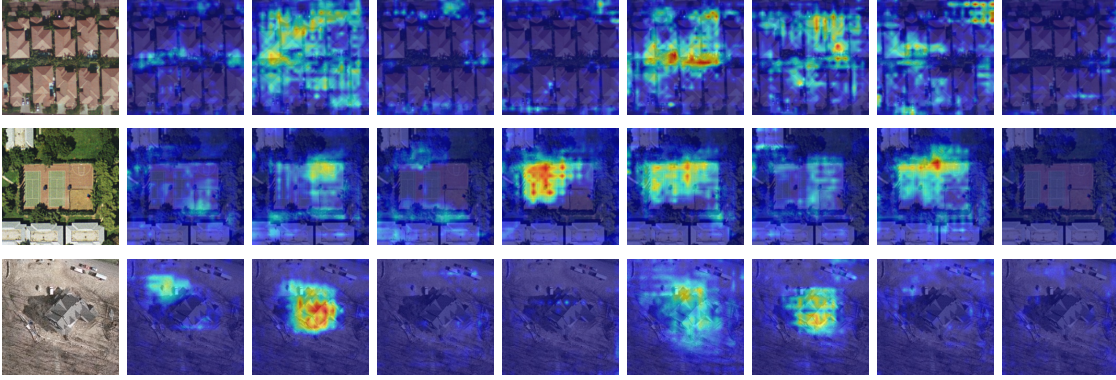


Figure 5.4: Example class attention maps of (a) images in UCM multi-label dataset with respect to (b) bare soil, (c) building, (d) car, (e) court, (f) grass, (g) pavement, (h) tree, and (i) water. Red indicates strong activations, while blue represents non-activations. Besides, normalization is performed based on each row for a fair comparison among class attention maps of the same images.

5.2 Reasoning about label relations for multi-label object classification

5.2.1 Motivation

In order to explicitly model label relations, we propose a label relational inference network for multi-label aerial image classification. This work is inspired by recent successes of relation networks in visual question answering [212], object detection [213], video classification [214], activity recognition in videos [215], and semantic segmentation [216]. A relation network is characterized by its inherent capability of inferring relations between an individual entity (e.g., a region in an image or a frame in a video) and all other entities (e.g., all regions in the image or all frames in the video). Besides, to increase the effectiveness of relational reasoning, we make use of a spatial transformer, which is often used to enhance the transformation invariance of deep neural networks [217], to reduce the impact of irrelevant semantic features.

5.2.2 Methodology

As illustrated in Figure 5.5, the proposed network comprises three components: a label-wise feature parcel learning module, an attentional region extraction module, and a label relational inference module. Let L be the number of object labels and l be the l -th label. The label-wise feature parcel learning module is designed to extract high-level feature maps \mathbf{X}_l with K channels, termed as *feature parcel*, for each label l . The attentional region extraction module is used to localize discriminative regions in each \mathbf{X}_l and generate an attentional feature parcel \mathbf{A}_l , which is supposed to contain the most relevant semantics with respect to the label l . Finally, relations among \mathbf{A}_l and all other label-wise attentional feature parcels are reasoned about by the label relational inference module for predicting the presence of the object l .

Label-wise Feature Parcel Learning. We take a standard CNN as the backbone of the label-wise feature parcel learning module in our model. As shown in Figure 5.5, an

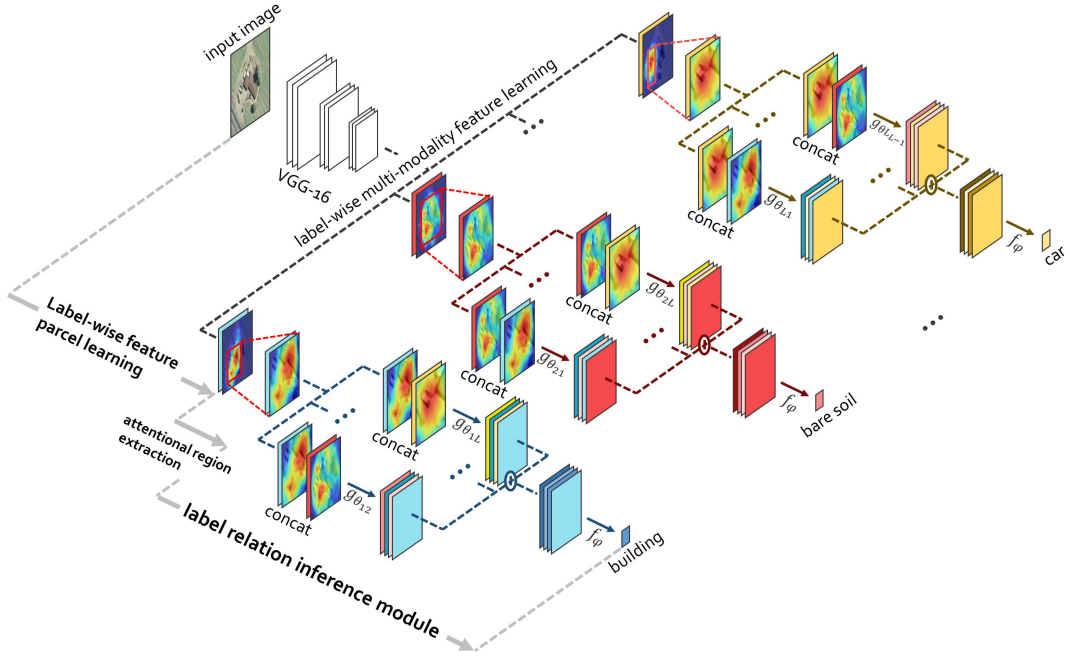


Figure 5.5: The architecture of the proposed attention-aware label relational reasoning network.

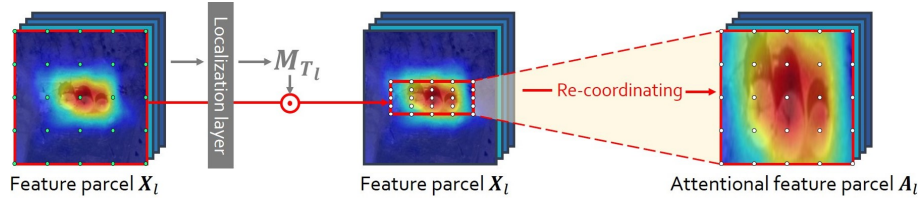


Figure 5.6: Illustration of the attentional region extraction module. Green dots in the left image indicate the feature parcel grid G_{X_l} . White dots in the middle image represent the attentional feature parcel grid $G_{X_l^{attn}}$, while those in the right image indicate re-coordinated $G_{X_l^{attn}}$. Notably, the structure of re-coordinated $G_{X_l^{attn}}$ is identical to that of G_{X_l} , and values of pixels located at grid points in re-coordinated $G_{X_l^{attn}}$ are obtained from those in $G_{X_l^{attn}}$. For example, the pixel at the left top corner grid point in re-coordinated $G_{X_l^{attn}}$ is assigned with the value of that at the left top corner of $G_{X_l^{attn}}$.

airial image is first fed into a CNN (e.g., VGG-16), which consists of only convolutional and max-pooling layers, for generating high-level feature maps. Subsequently, these features are encoded into L feature parcels for each label l via a label-wise multi-modality feature learning layer. To implement this layer, we first employ a convolutional layer with KL filters, whose size is 1×1 , to extract KL feature maps. Afterwards, we divide these features into L feature parcels, and each includes K feature maps. That is to say, for each label, K specific feature maps are learned, so-called *feature parcel*, to extract discriminative semantics after the end-to-end training of the whole network. We denote the feature parcel for label l as X_l in the following statements.

In our experiment, we notice that X_l with a higher resolution is beneficial for the subsequent module to localize discriminative regions, as more spatial contextual cues are

included. Accordingly, we discard the last max-pooling layer in VGG-16, leading to a spatial size of 14×14 for outputs. Weights are initialized with pre-trained VGG-16 on ImageNet but updated during the training phase.

Attentional Region Extraction Module. Although label-wise feature parcels can be directly applied to exploring label dependencies [141], less informative regions (see blue areas in Figure 5.6) may bring noise and further reduce the effectiveness of these feature parcels. As shown in the left image of Figure 5.6, weakly activated regions indicate a loose relevance to the corresponding label, while highlighted regions suggest a strong region-label relevance. To diminish the influence of unrelated regions, we employ an attentional region extraction module to automatically extract discriminative regions from label-wise feature parcels.

We localize and re-coordinate attentional regions from \mathbf{X}_l with a learnable spatial transformer. Particularly, we sample a feature parcel \mathbf{X}_l into a regular spatial grid $G_{\mathbf{X}_l}$ (cf. green dots in the left image of Figure 5.6) according to the spatial resolution of \mathbf{X}_l and regard pixels in \mathbf{X}_l as points on the grid $G_{\mathbf{X}_l}$ with coordinates (x_l, y_l) . Similarly, we can define coordinates of a new grid, attentional region grid $G_{\mathbf{X}_l^{attn}}$ (see white dots in the middle image of Figure 5.6), as (x_l^{attn}, y_l^{attn}) , and the number of grid points along with the height and width is equivalent to that of $G_{\mathbf{X}_l}$. As demonstrated in [217] that $G_{\mathbf{X}_l^{attn}}$ can be learned by performing spatial transformation on $G_{\mathbf{X}_l}$, (x_l^{attn}, y_l^{attn}) can be calculated with the following equation:

$$\begin{bmatrix} x_l^{attn} \\ y_l^{attn} \end{bmatrix} = \mathbf{M}_{T_l} \begin{bmatrix} x_l \\ y_l \\ 1 \end{bmatrix}, \quad (5.10)$$

where \mathbf{M}_{T_l} is a learnable transformation matrix, and grid coordinates, x_l and y_l , are normalized to $[-1, 1]$. Considering that this module is designed for localization, we only adopt scaling and translation in our case. Hence Eq. 5.10 can be rewritten as

$$\begin{bmatrix} x_l^{attn} \\ y_l^{attn} \end{bmatrix} = \begin{bmatrix} s_{x_l} & 0 & t_{x_l} \\ 0 & s_{y_l} & t_{y_l} \end{bmatrix} \begin{bmatrix} x_l \\ y_l \\ 1 \end{bmatrix}, \quad (5.11)$$

where s_{x_l} and s_{y_l} indicate scaling factors along x- and y-axis, respectively, and t_{x_l} and t_{y_l} represent how feature maps should be translated along both axes. Notably, since different objects distribute variously in aerial images, \mathbf{M}_{T_l} is learned for each object label l individually. In other words, extracted attentional regions are label-specific and capable of improving the effectiveness of label-wise features.

As to the implementation of this module, we first vectorize \mathbf{X}_l with a flatten function and then employ a localization layer (e.g., a fully connected layer) to estimate elements in \mathbf{M}_{T_l} from the vectorized \mathbf{X}_l . Afterwards, attentional region grid coordinates (x_l^{attn}, y_l^{attn}) can be learned from (x_l, y_l) with Eq. 5.11, and values of pixels at (x_l^{attn}, y_l^{attn}) is able to be obtained from neighboring pixels by bilinear interpolation. Finally, the attentional region grid $G_{\mathbf{X}_l^{attn}}$ is re-coordinated to a regular spatial grid, which shares an identical structure with $G_{\mathbf{X}_l}$, for yielding the final attentional feature parcel \mathbf{A}_l .

Label Relational Inference Module. Being the core of our model, the label relational inference module is designed to fully exploit label interrelations for inferring existences of all labels. Before diving into this module, we define the pairwise label relation as a composite function with the following equation:

$$\text{LR}(\mathbf{A}_l, \mathbf{A}_m) = f_\phi(g_{\theta_{lm}}(\mathbf{A}_l, \mathbf{A}_m)), \quad (5.12)$$

5.2 Reasoning about label relations for multi-label object classification

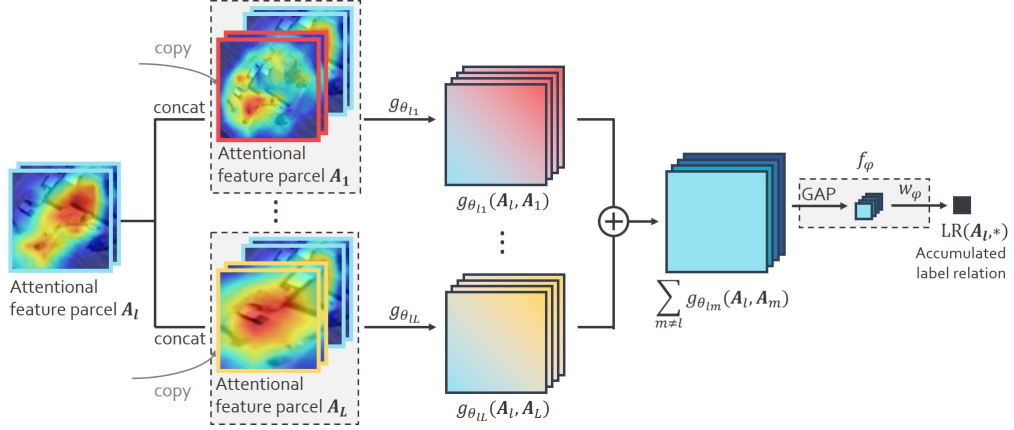


Figure 5.7: Illustration of the label relation module.

where the input is a pair of attentional feature parcels, \mathbf{A}_l and \mathbf{A}_m , and l and m range from 1 to L . The functions $g_{\theta_{lm}}$ and f_ϕ are used to reason about the pairwise relation between label l and m . More specifically, the role of $g_{\theta_{lm}}$ is to reason about whether there exist relations between the two objects and how they are related. In previous works [212, 215], a Multilayer Perceptron (MLP) is commonly employed as $g_{\theta_{lm}}$ for its simplicity. However, spatial contextual semantics are not taken into account in this way. To address such issue, here, we make use of 1×1 convolution instead of an MLP to explore spatial information. Furthermore, f_ϕ is applied to encode the output of $g_{\theta_{lm}}$ into the final pairwise label relation $\text{LR}(\mathbf{A}_l, \mathbf{A}_m)$. In our case, f_ϕ consists of a global average pooling layer and an MLP, which finally yields the relation between label l and m .

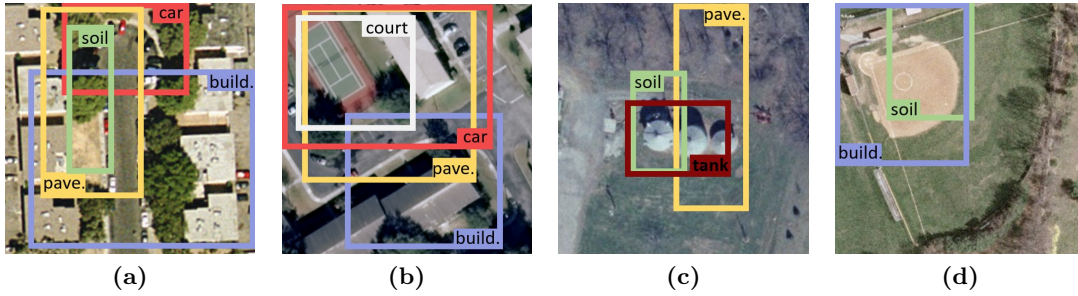
Following the motivation of our work, we infer each label by accumulating all related pairwise label relations, and the accumulated label relation for object label l is defined as:

$$\text{LR}(\mathbf{A}_l, *) = f_\phi\left(\sum_{m \neq l} g_{\theta_{lm}}(\mathbf{A}_l, \mathbf{A}_m)\right), \quad (5.13)$$

where $*$ represents all attentional feature parcels except \mathbf{A}_l . Based on this formula, we implement the label relational inference module with the following steps (taking the prediction of label l as an example): 1) \mathbf{A}_l and every other attentional feature parcel are concatenated and fed into a 1×1 convolutional layer, respectively. 2) Afterwards, a global average pooling layer is employed to transform $g_{\theta_{lm}}(\mathbf{A}_l, \mathbf{A}_m)$ into vectors, which are then element-wise added. 3) Finally, the output is fed into an MLP layer with trainable parameters ϕ to produce the accumulated label relation $\text{LR}(\mathbf{A}_l, *)$. Note that $g_{\theta_{lm}}$ is a learnable unit, which models pairwise relations using convolutions. Through the end-to-end training, it could be expected to learn data-driven label relations. Experiments have verified that learned label relations are in line with prior knowledge. Since we expect the model to predict probabilities, an activation function σ is utilized to restrict each output digit to $[0, 1]$. For label l , a digit approaching 1 implies a high probability of its presence, while one closing 0 suggests the absence. Figure 5.7 presents an visual illustration of the label relational inference module.

Table 5.2: Comparisons of the classification performance on UCM Multi-label Dataset (%).

Network	mEF ₁	mEF ₂	mEP	mER	mCP	mCR
VGGNet [10]	78.54	80.17	79.06	82.30	86.02	80.21
VGG-RBFNN [126]	78.80	81.14	78.18	83.91	81.90	82.63
CA-VGG-BiLSTM [141]	79.78	81.69	79.33	83.99	85.28	76.52
AL-RN-VGGNet	85.70	85.81	87.62	86.41	91.04	81.71
GoogLeNet [12]	80.68	82.32	80.51	84.27	87.51	80.85
GoogLeNet-RBFNN [126]	81.54	84.05	79.95	86.75	86.19	84.92
CA-GoogLeNet-BiLSTM [141]	81.82	84.41	79.91	87.06	86.29	84.38
AL-RN-GoogLeNet	85.24	85.33	87.18	85.86	91.03	81.64
ResNet-50 [15]	79.68	80.58	80.86	81.95	88.78	78.98
ResNet-RBFNN [126]	80.58	82.47	79.92	84.59	86.21	83.72
CA-ResNet-BiLSTM [141]	81.47	85.27	77.94	89.02	86.12	84.26
AL-RN-ResNet	86.76	86.67	88.81	87.07	92.33	85.95

**Figure 5.8:** Example attentional regions for car, bare soil (soil), building (build.), pavement (pave.), court, and tank in various scenes (a)-(d) in the UCM multi-label dataset.

5.2.3 Results

Table 5.2 exhibits experimental results on the UCM multi-label dataset. We can observe that our model surpasses all competitors on the UCM multi-label dataset with variant backbones. Specifically, AL-RN-VGGNet increases mean F_1 and F_2 scores by 7.16% and 5.64%, respectively, in comparison with VGGNet. Compared to CA-VGG-BiLSTM, which resorts to employing a bidirectional LSTM structure for exploring label dependencies, our network obtains an improvement of 5.92% in the mean F_1 score. To further evaluate the proposed network, we visualize attentional regions learned from the second module. Figure 5.8 shows some examples of learned attentional regions. As we can see, most attentional regions concentrate on areas covering objects of interest.

We also evaluate our network on the AID multi-label dataset which is produced by relabeling 3000 aerial images in the AID dataset. For each of the 30 scene categories in AID, we evenly select 100 images and assign each multiple labels through manual visual inspection. For more numerical and visual results, please refer to **Appendix B**.

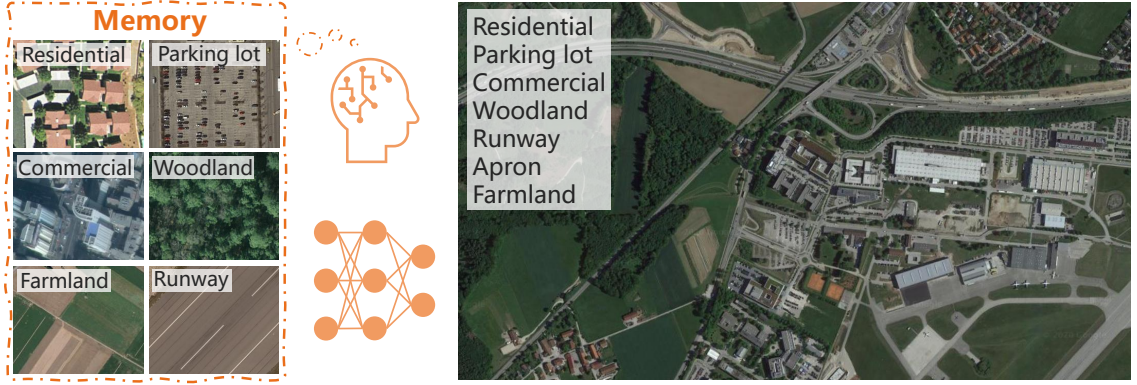


Figure 5.9: Illustration of how humans learn to perceive unconstrained aerial images being composed of multiple scenes. We first learn and memorize individual aerial scenes. Then we can possess the capability of understanding complex scenarios by learning from only a limited number of hard instances. We believe by simulating this learning process, a deep neural network can also learn to interpret multi-scene aerial images.

5.3 Memorizing scene prototypes for multi-scene recognition

5.3.1 Motivation

To learn networks for multi-scene recognition, huge quantities of well-annotated multi-scene images are needed. However, we note that such annotations are not easy in the remote sensing community. To solve such a limitation, we propose to train a network with only a small number of labeled multi-scene images but a huge amount of existing, annotated single-scene data. Our motivation is based on an intuitive observation about how humans learn to perceive complex scenes being composed of multiple entities [218, 219, 220]: we first learn and memorize individual objects (through flash cards for example) when we were babies and then possess the capability of understanding complex scenarios by learning from only a limited number of hard instances (cf. Figure 5.9). We believe that this learning process also applies to the interpretation of multi-scene aerial images. Driven by this observation, we propose a novel network, termed as prototype-based memory network (PM-Net).

5.3.2 Methodology

As shown in Figure 5.10, the proposed PM-Net consists of three essential components: a prototype learning module, an external memory, and a memory retrieval module. Specifically, the prototype learning module is devised to encode prototype representations of aerial scenes, which are then stored in the external memory. The memory retrieval module is responsible for retrieving scene prototypes related to query images through a multi-head attention mechanism. Eventually, retrieved scene prototypes are utilized to infer the existence of multiple scenes in the query image.

Scene Prototype Learning and Writing. Following the observation, we propose to learn and memorize scene prototypes with the support of single-scene aerial images. The procedure consists of two stages. We first employ an embedding function to learn semantic

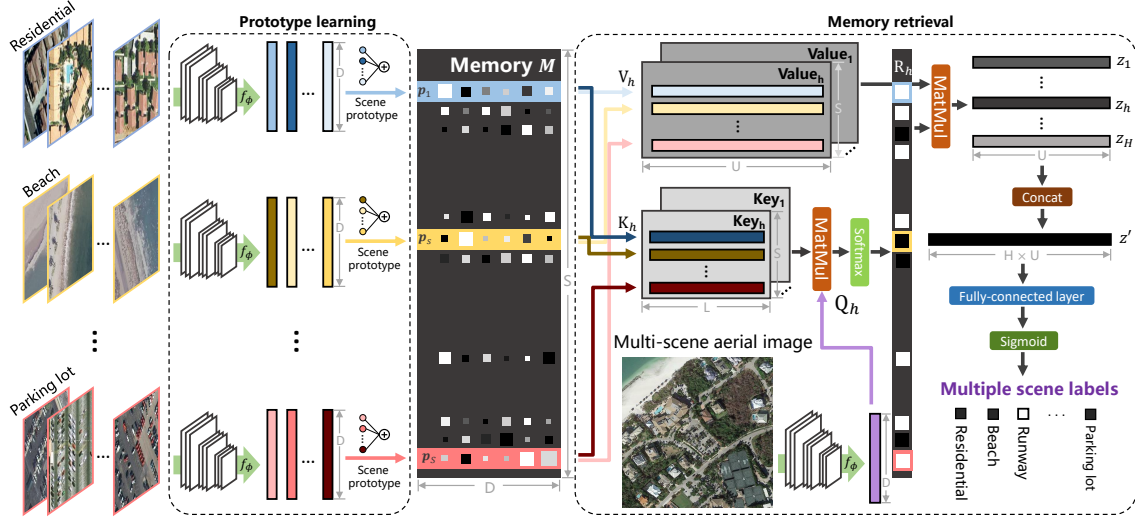


Figure 5.10: Architecture of the proposed PM-Net. Particularly, we first learn scene prototypes p_s from well-annotated single-scene aerial images and then store them in the external memory M of PM-Net. Afterwards, given a query multi-scene image, a multi-head attention-based memory retrieval module is devised to retrieve scene prototypes that are relevant to the query image, yielding z' for the prediction of multiple labels. f_ϕ denotes the embedding function, and its output is a D -dimensional feature vector. S and H represent numbers of scenes and heads, respectively. L and U denote channel dimensions of the key and value in the memory retrieval module.

representations of all single-scene images. Then, feature representations belonging to the same scene category are encoded into a scene prototype and stored in the external memory.

Formally, let \mathbf{X}_i^s denote the i -th single-scene image belonging to scene s , and i ranges from 1 to N_s . N_s is the number of samples annotated as s . The embedding function f_ϕ can be learned via the following objective function:

$$\mathcal{L}(\mathbf{X}_i^s, \mathbf{y}^s) = -\mathbf{y}^s \log \frac{\exp(-g_\theta(f_\phi(\mathbf{X}_i^s)))}{\sum_s \sum_i \exp(-g_\theta(f_\phi(\mathbf{X}_i^s)))}, \quad (5.14)$$

where ϕ represents learnable parameters of f_ϕ , and \mathbf{y}^s is a one-hot vector denoting the scene label of \mathbf{X}_i^s . g_θ is a multilayer perceptron (MLP) with parameters θ and its outputs are activated by a softmax function to predict probability distributions. Following the overwhelming trend of deep learning, here we employ a deep CNN, e.g., ResNet-50 [15], as the embedding function f_ϕ and learn its parameters on public single-scene aerial image datasets. After sufficient training, f_ϕ is expected to be capable of learning discriminative representations for different aerial scenes.

Once f_ϕ is learned, the scene prototype can be computed by averaging representations of all aerial images belonging to the same scene [221, 222, 223]. Let p_s be the prototype representation of scene s . We calculate p_s with the following equation:

$$p_s = \frac{1}{N_s} \sum_{i=1}^{N_s} f_\phi(\mathbf{X}_i^s). \quad (5.15)$$

By doing so, in the single-scene classification, an image closely around p_s in the common embedding space is supposed to belong to scene s . Similarly, in the multi-scene scenario,

the representation of an aerial image comprising scene s should show high relevance with \mathbf{p}_s . After encoding all scene prototypes, the external memory \mathbf{M} can be formulated as follows:

$$\mathbf{M} = [\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_S]^T, \quad (5.16)$$

where S denotes the number of scenes. $[\dots, \dots]$ represents the concatenation operation. Given that \mathbf{p}_s is a D -dimensional vector, \mathbf{M} is a matrix of $S \times D$. Note that D varies when using different backbone CNNs as embedding functions.

Multi-head Attention-based Memory Retrieval. Inspired by successes of the multi-head self-attention mechanism [109] in natural language processing tasks [224, 225, 226, 227], we develop a multi-head attention-based memory retrieval module to retrieve scene prototypes from the memory \mathbf{M} for a given query image \mathbf{X} . Given a query multi-scene aerial image \mathbf{X} , to retrieve relevant scene prototypes from \mathbf{M} , we develop a multi-head attention-based memory retrieval module. In particular, we first extract the feature representation of \mathbf{X} through the same embedding function f_ϕ and linearly project it to an L -dimensional query $\mathbf{Q}(\mathbf{X})$. Similarly, we transform the external memory \mathbf{M} into key $\mathbf{K}(\mathbf{M})$ and value $\mathbf{V}(\mathbf{M})$, and both are implemented as MLPs. The channel dimension of the key is L , while that of the value is U . The relevance between \mathbf{X} and each scene prototype \mathbf{p}_s can be measured by dot product similarity and a softmax function as follows:

$$\mathbf{R}(\mathbf{X}, \mathbf{M}) = \text{softmax}\left(\frac{\mathbf{Q}(f_\phi(\mathbf{X})) \cdot \mathbf{K}(\mathbf{M})^T}{\sqrt{L}}\right). \quad (5.17)$$

The output is an S -dimensional vector, where each component represents a relevance probability that a specific scene prototype is related to the query image. Subsequently, the retrieved scene prototypes are computed by weight-summing all values with the following equation:

$$\mathbf{z} = \mathbf{R}(\mathbf{X}, \mathbf{M}) \cdot \mathbf{V}(\mathbf{M}). \quad (5.18)$$

Since the memory retrieval is designed in a multi-head fashion, the final retrieved prototype is reformulated as follows:

$$\mathbf{z}' = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_H], \quad (5.19)$$

where H denotes the number of heads, and each head yields a retrieved prototype \mathbf{z}_h by transforming \mathbf{X} and \mathbf{M} to the variant query $\mathbf{Q}_h(f_\phi(\mathbf{X}))$, key $\mathbf{K}_h(\mathbf{M})$, and value $\mathbf{V}_h(\mathbf{M})$. Eventually, the output \mathbf{z}' is fed into a fully-connected layer followed by a sigmoid function for inferring presences of aerial scenes.

5.3.3 Results

In order to widely evaluate the performance of our method, we design a dataset configuration, UCM2MAI, based on common scene categories shared by UCM and MAI. Specifically, the UCM2MAI configuration consists of 1600 single-scene aerial images from the UCM dataset and 1649 multi-scene images from our MAI dataset.

For a comprehensive evaluation, we compare the proposed PM-Net with two baselines, CNN* and CNN. The former is initialized with parameters pretrained on ImageNet, and the latter is pretrained on single-scene datasets. Besides, we compare our network with a memory network, Mem-N2N [228] and a K-Branch CNN [137]. Since Mem-N2N was proposed for the question answering task, we adapt it to our task by replacing its inputs,

Table 5.3: Numerical Results on UCM2MAI (%).

Model	mEF ₁	mEF ₂	mEP	mER	mCP	mCR
VGGNet* [10]	32.16	32.79	35.08	34.35	21.74	22.57
VGGNet [10]	51.42	49.04	62.00	48.38	36.80	27.44
Mem-N2N-VGGNet [228]	52.16	50.93	57.26	50.73	20.79	22.58
K-Branch CNN [137]	47.04	43.15	64.57	41.83	37.93	22.28
proposed PM-VGGNet	54.42	51.16	67.35	49.95	47.24	26.79
Inception-V3* [12]	48.03	44.37	62.22	42.80	47.36	20.43
Inception-V3 [12]	53.96	51.28	65.47	50.49	51.03	32.88
Mem-N2N-Inception-V3 [228]	56.06	55.27	62.95	55.92	47.90	30.48
proposed PM-Inception-V3	58.56	58.06	64.17	58.73	46.44	26.47
ResNet* [15]	48.36	45.00	63.90	43.84	53.63	28.35
ResNet [15]	51.39	48.31	65.33	47.37	51.89	30.54
Mem-N2N-ResNet [228]	54.31	51.45	63.97	50.31	44.33	24.58
proposed PM-ResNet	56.89	54.11	69.85	53.38	55.93	29.76
NASNet* [24]	43.64	39.94	58.56	38.39	46.01	19.69
NASNet [24]	52.03	49.43	64.24	48.75	49.99	33.75
Mem-N2N-NASNet [228]	55.17	53.05	64.71	52.65	49.60	29.14
proposed PM-NASNet	60.13	59.57	67.04	60.42	58.60	35.04

CNN* is initialized with weights pretrained on ImageNet.

CNN, Mem-N2N, and PM-Net are initialized with parameters pretrained on the UCM dataset.

i.e., embeddings of *questions* and *statements*, with *query image representations* $f_\phi(\mathbf{X})$ and *scene prototypes* \mathbf{p}_s , respectively. To be more specific, we feed \mathbf{X} to a CNN backbone and take its output as the input of Mem-N2N. Scene prototypes are stored in the memory of Mem-N2N and retrieved according to $f_\phi(\mathbf{X})$. The initialization of f_ϕ is the same as that of our network, and the entire Mem-N2N is trained in an end-to-end manner. Various backbones of embedding functions are test, and quantitative results are reported in Table 5.3. Furthermore, we visualize features of single-scene images learned by VGGNet on UCM and AID datasets via t-SNE, respectively. As shown in Figure 5.11, extracted features are discriminative and separable in the embedding space, which demonstrates the effectiveness of learning the embedding function on single-scene aerial image datasets.

We also evaluate the effectiveness of PM-Net on the AID dataset by constructing AID2MAI which is composed of 7050 and 3239 aerial images from the AID and MAI datasets, respectively. For more experimental results and technical details, please refer to **Appendix C**.

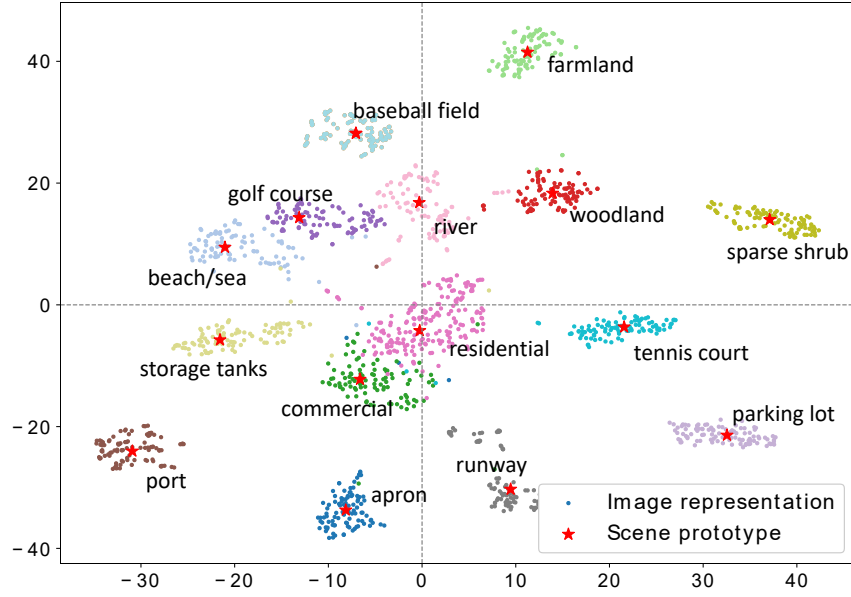


Figure 5.11: T-SNE visualization of image representations and scene prototypes learned by VG-Net on the UCM dataset. Dots in the same color represent features of images belonging to the same scene, and stars denote scene prototypes.

5.4 A large-scale dataset and benchmark for multi-scene recognition

5.4.1 Motivation

Multi-scene recognition in single aerial images is a more realistic yet challenging problem, and it refers to assigning multiple scene labels to an aerial image with no constraints, such as centering dominant scenes and eliminating clutter scenes. Compared to the conventional scene recognition task, multi-scene recognition is more arduous because 1) images are large-scale and unconstrained, and 2) all present scenes in an aerial image need to be exhaustively recognized. Figure 5.12(b) shows an example of multi-scene aerial image and corresponding multiple scene-level labels. We can see that not only dominant scenes (e.g., residential and woodland) but also trivial scenes (e.g., bridge and parking lot) are annotated, which draws a more comprehensive picture for the unconstrained image.

5.4.2 Benchmark

We collect 100,000 high-resolution aerial images from Google Earth imagery, which cover six continents, Europe, Asia, North America, South America, Africa, and Oceania, and eleven countries including Germany, France, Italy, England, Spain, Poland, Japan, the United States, Brazil, South Africa, and Australia (cf. Figure 5.14). This can ensure high intra-class diversity, as different scene appearances resulted from different cultural regions are covered. The spatial resolution of each image ranges from 0.3 m/pixel to 0.6 m/pixel, and the spatial size of images is 512×512 pixels. In contrast to single-scene image datasets [46, 173, 34, 45], we put no constraints on the location and area of the dominant/trivial scene in an image during the data collection process. Some example multi-scene images are exhibited in Figure 5.13. In total, 36 scene categories are defined:

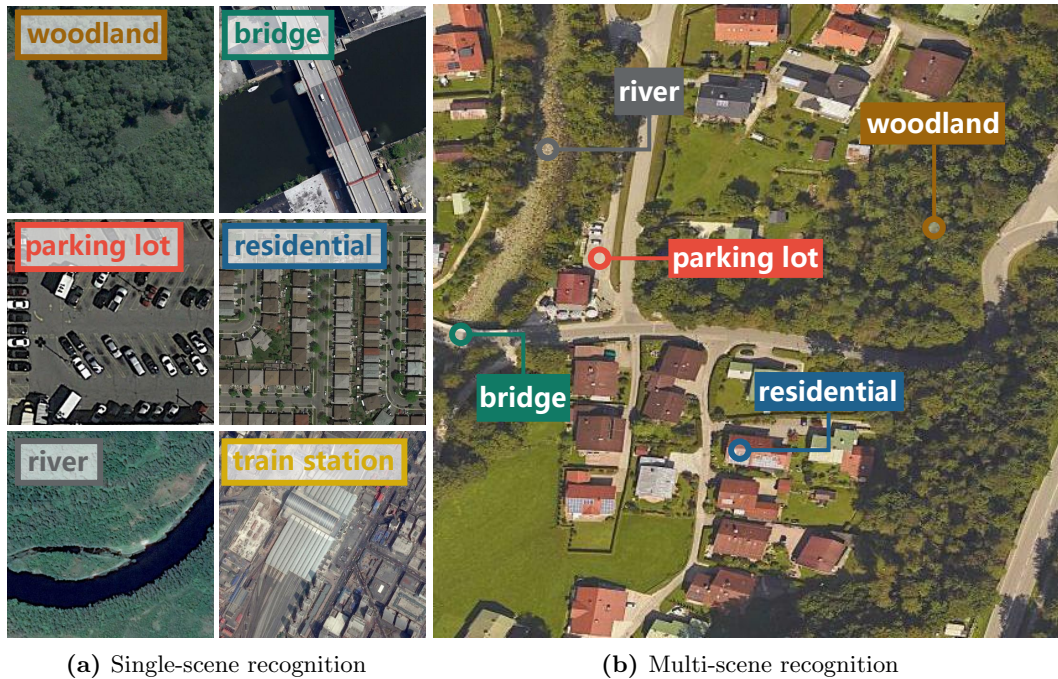


Figure 5.12: Examples of images utilized in (a) single-scene and (b) multi-scene recognition tasks. In (a), each aerial image is assigned one scene label, while in (b), labels of all present scenes are inferred. In comparison with (b), (a) might suffer from partial scene understanding, as only one label is predicted even if there indeed exist multiple scenes in an image. For a clear visualization, locations of scenes are marked in (b).

apron, baseball field, basketball field, beach, bridge, cemetery, commercial, farmland, woodland, golf course, greenhouse, helipad, lake/pond, oil field, orchard, parking lot, park, pier, port, quarry, railway, residential, river, roundabout, runway, soccer field, solar farm, sparse shrub, stadium, storage tanks, tennis court, train station, wastewater, plant, wind turbine, works, and sea.

To obtain crowdsourced annotations, we first localize each image in OSM with coordinates of its four corners. Afterwards, we parse properties of scenes present in the corresponding region and label images accordingly. In this way, crowdsourced annotations of all aerial images can be automatically yielded at a very low cost compared to conventional manual labeling. However, these almost free annotations might suffer from noise, and the performance of networks directly trained on them could be degraded. Therefore, we visually inspect 14,000 images from all six continents and correct their labels, yielding a subset, MultiScene-Clean. Figure 5.14 shows the coordinate distribution of all images, and the number of samples associated with each scene is present in Figure 5.15. Compared to other scene recognition datasets introduced in Chapter 3.4, our dataset is featured by its manifold labels per image and the available crowdsourced annotations. Figure 5.16 further shows the number of images associated with different numbers of scenes.

Compared to existing aerial scene datasets, our dataset brings more challenges to the field of scene interpretation from the following three perspectives:

- Images are unconstrained and large-scale, and thus scenes are likely to be incomplete and trivial, which makes recognition more difficult.

5.4 A large-scale dataset and benchmark for multi-scene recognition

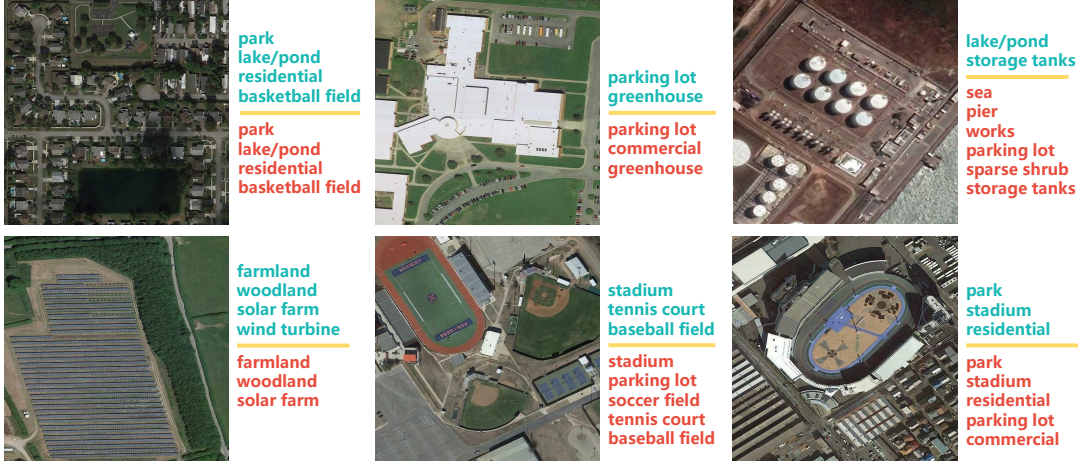


Figure 5.13: Example multi-scene aerial images with their crowdsourced and clean annotations in the MultiScene dataset.

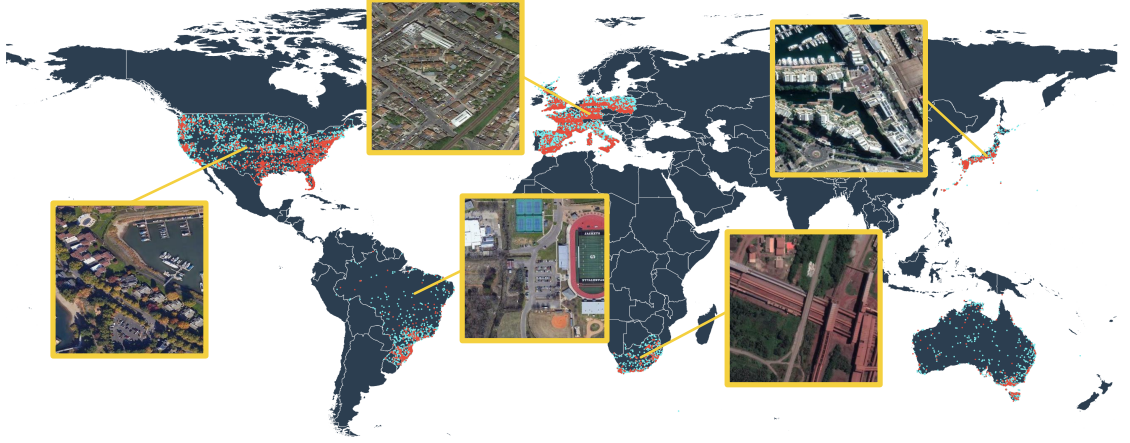


Figure 5.14: Coordinate distributions and examples of multi-scene aerial images in our dataset. Red dots denote images with both crowdsourced and clean labels, and cyan dots represent images with only crowdsourced scene labels.

- The long-tail sample distribution (see Figure 5.15) poses a challenge of learning unbiased models on an imbalanced dataset.
- We gather images from different cultural regions, which results in a high intra-class variation.

5.4.3 Results

Since MultiScene allows researches in not only recognizing aerial scenes in the wild but also learning from noisy crowdsourced labels, we assess all baselines with respect to both tasks.

Multi-scene Recognition with Cleanly-labeled Data. To evaluate baselines for our task, we conduct experiments on the MultiScene-Clean dataset and report quantitative results in Table 5.4. It can be seen that ResNeXt-101 achieves the best mAP (64.8%),

5 Summary of works

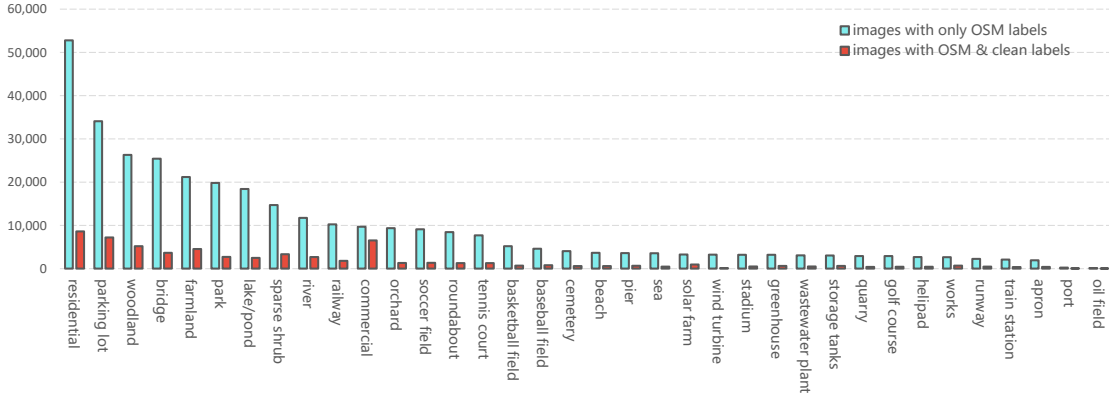


Figure 5.15: Sample distributions of all scene categories in our dataset. Each cyan bar indicates the number of images assigned only OSM labels with respect to each scene category, and red bars represent numbers of images with both OSM and clean labels.

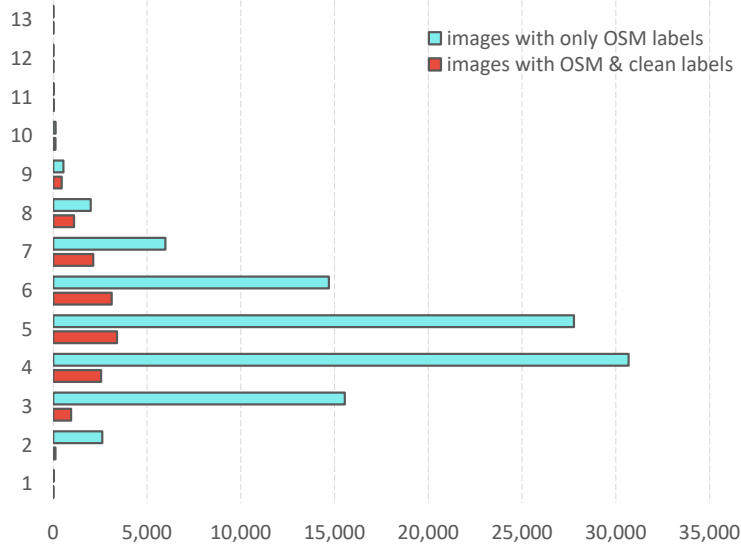


Figure 5.16: The number of images associated with different numbers of scenes. Y-axis indicates the number of scenes, and X-axis represents the number of images. The legend is the same as that in Figure 5.15.

mEF₁ (70.2%), and OF₁ score (71.3%), which demonstrate its high performance and robustness in this task from almost all perspectives. LR-ResNet-50 gains the highest value in mCF₁ (59.0%) owing to its capability of reasoning about relations among various scenes. Moreover, such a reasoning capability also enables LR-ResNet-50 to surpass the other baselines in all recall metrics, as scenes tend to be predicted as positive once its related scenes are recognized. Another observation is that MnasNet, SqueezeNet, and ShuffleNet-V2 show relatively poor performance due to their light-weight designs. Compared to deep neural networks, traditional machine learning algorithms achieve lower scores in all metrics.

Learning from Noisy Crowdsourced Labels. We investigate networks learned from noisy crowdsourced labels for our task on the MultiScene dataset. To ensure a fair comparison, we utilize the same test set as the previous experiment and compare the

5.4 A large-scale dataset and benchmark for multi-scene recognition

Table 5.4: Numerical results of baseline models on the MultiScene-Clean dataset (%). Models are trained and tested on cleanly-labeled images, and the best scores are shown in bold.

Model	mAP	mCP	mCR	mCF ₁	mEP	mER	mEF ₁	OP	OR	OF ₁
SVM	14.9	19.6	8.4	8.6	62.2	32.8	41.1	66.9	32.2	43.5
RF	15.6	25.4	8.7	9.5	64.6	32.5	41.4	70.9	32.1	44.2
XGBOOST	16.9	34.1	11.2	12.8	67.0	37.4	45.8	69.6	36.5	47.9
VGG-16	56.5	63.3	47.9	53.6	74.9	64.3	67.0	73.6	63.1	67.9
VGG-19	56.4	62.9	47.7	53.3	74.8	64.1	66.8	73.5	62.7	67.7
Inception-V3	53.5	65.0	40.8	48.5	74.2	59.9	63.9	73.0	58.6	65.0
ResNet-50	62.0	74.8	45.9	55.1	79.7	62.7	67.9	79.0	61.4	69.1
ResNet-101	63.0	75.9	46.6	55.8	79.9	64.3	69.1	79.2	63.1	70.3
ResNet-152	63.8	74.9	49.1	57.7	80.8	64.0	69.2	80.1	62.8	70.4
SqueezeNet	46.3	58.1	36.8	43.5	71.3	58.0	61.3	70.0	56.9	62.7
MobileNet-V2	58.8	70.9	44.8	53.1	77.6	62.7	67.0	76.6	61.6	68.3
ShuffleNet-V2	50.7	61.8	38.1	45.7	73.8	58.2	62.5	73.0	57.0	64.0
DenseNet-121	62.2	74.6	45.1	54.4	79.5	61.8	67.3	79.1	60.6	68.6
DenseNet-169	63.2	76.7	45.8	55.3	80.4	63.4	68.6	79.6	62.3	69.9
ResNeXt-50	63.4	77.3	45.0	54.2	78.5	64.3	68.6	77.8	63.2	69.8
ResNeXt-101	64.8	76.5	48.6	57.3	79.3	66.6	70.2	78.5	65.4	71.3
MnasNet	53.8	61.8	42.9	49.9	73.0	59.4	63.0	72.1	58.1	64.3
KFBNet	58.8	68.8	45.2	53.3	77.9	64.2	68.1	77.3	63.0	69.4
FACNN	56.5	60.3	48.7	52.6	73.1	65.3	66.8	71.6	64.1	67.7
SAFF	61.8	72.5	48.1	56.7	79.4	63.9	68.6	78.7	62.8	69.9
LR-VGG-16	58.1	67.7	46.7	54.2	77.3	64.6	68.0	76.2	63.5	69.2
LR-ResNet-50	63.1	68.1	53.1	59.0	76.7	67.6	69.7	75.3	66.5	70.6

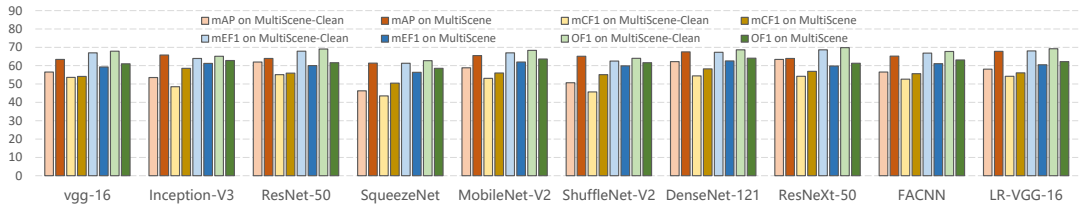


Figure 5.17: Comparisons of the performance of networks trained on images with clean (light-color bars) and crowdsourced (dark-color bars) annotations, respectively. For each network, the left four bars represent class-based scores, mAPs and CF₁, while the right four bars indicate EF₁ and OF₁ scores.

performance of several networks trained on MultiScene-Clean and MultiScene datasets in Figure 5.17. It can be observed that higher class-based scores (see orange and brown bars in Figure 5.17) are obtained when using massive crowdsourced labels. That is to say, although crowdsourced labels influence the overall performance of networks, comparisons in class-based scores also suggest their great potential.

For more experimental results and technical details, please refer to **Appendix D**.

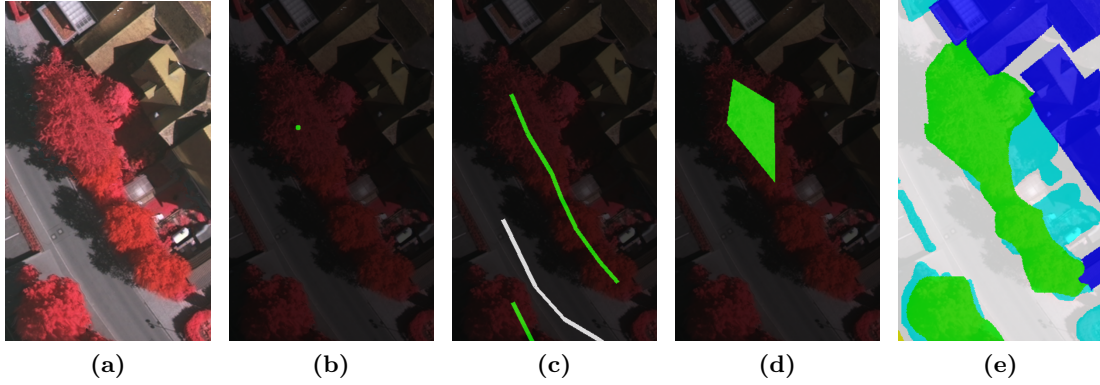


Figure 5.18: Comparisons of different levels of scribbled annotations. Trees (marked as green) are taken as an example here. Images from left to right are (a) an aerial image, (b) point-, (c) line- and (d) polygon-level scribbled annotations, and (e) dense pixel-wise labels.

5.5 Semantic segmentation of aerial imagery with sparse annotations

5.5.1 Motivation

Training a fully supervised segmentation CNN requires a huge volume of dense pixel-level ground truths, which are labor- and time-consuming to generate. Besides, expert annotators might be needed for correctly identifying pixels located at object boundaries and ambiguous regions (e.g., shadows in Figure 5.18) which also contributes to the high cost of dense pixel-wise annotations. To alleviate such a burden, we propose a simple yet effective framework for semantic segmentation of remote sensing imagery with low-cost annotations.

5.5.2 Methodology

Supervision with Sparse Annotations. Here we consider three levels of sparse annotations: point-, scribble-, and polygon-level. Specifically, point-level annotations indicate that, for an annotator interaction, only one single pixel is labeled. Scribble-level annotations, also called line-level annotations, are yielded by drawing a scribble line within an object and assigning all pixels along this line the same class label. Similarly, polygon-level annotations can be generated by drawing a polygon within an object and classifying pixels located in the polygon into the same semantic class. Examples of these three levels of annotations are shown in Figure 5.18.

To annotate large-scale images, we employ an online labeling platform, LabelMe ¹, and ask annotators to draw by following these rules: 1) for each class, annotations are supposed to cover diverse appearances (see region a, b, and c in Figure 5.19, where cars of different colors are annotated) and be located in different positions of the image separately. 2) polygon- and line-level annotations are not required to delineate object boundaries precisely, see the annotations of trees in Fig. 5.18c and 5.18d. In order to make the time

¹<http://labelme.csail.mit.edu/Release3.0/>

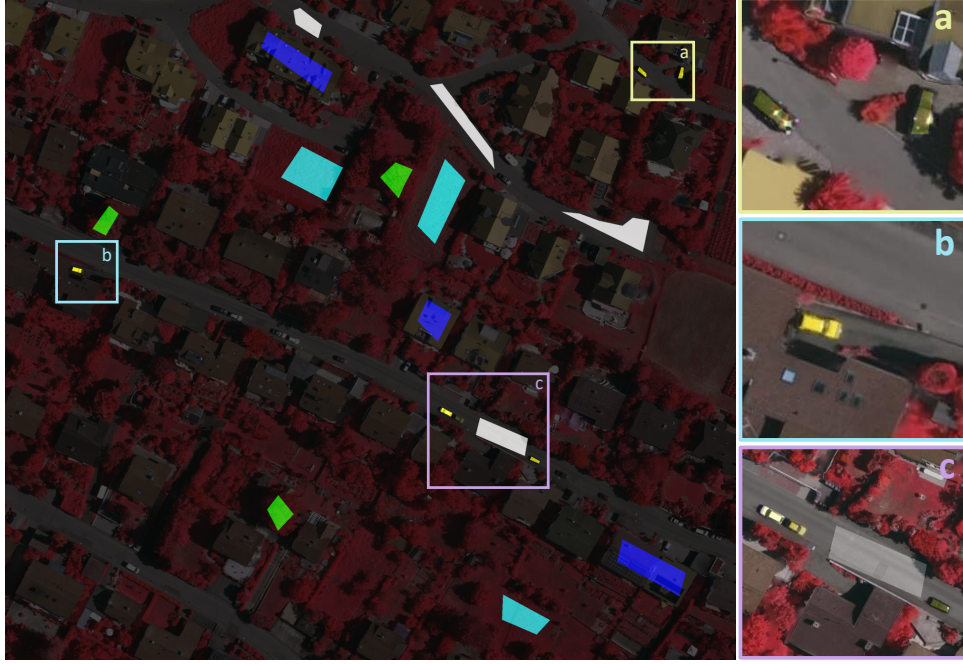


Figure 5.19: Example polygon-level annotations of an image (ID: 13) on the Vaihingen dataset. Annotations of cars are zoomed in to illustrate that annotations should include variant visual appearances for one class. Legend—white: impervious surfaces, blue: buildings, cyan: low vegetation, green: trees, yellow: cars.

spent on each level of scribbled annotations more equivalent, we ask 4 annotators (including 2 non-experts) to label 7, 5, and 3 objects per class for point-, line- and polygon-level annotations in each aerial image. As a consequence, sparse but accurate annotations can be provided rapidly without effort. Since a point- or line-level annotation is often located in the centre area of an object and distant from its boundary, we perform morphological dilation on all point- and line-level annotations with a disk of radius 3. Afterwards, pixels involved in dilated annotations are assigned the same class labels as their central points or lines. For polygon-level annotations, pixels within each polygon are assigned the corresponding classes. As to the labeling time, it took on average 133, 126, and 161 seconds per image to produce point-, line- and polygon-level annotations, respectively, for the Vaihingen dataset, and 177, 162, and 238 seconds per image for the Zurich Summer dataset.

Feature and Spatial Relational Regularization. When using sparse annotations, the vast majority of pixels in the training images are left unlabelled. In order to exploit both labeled and unlabeled pixels, we develop a semi-supervised methodology, named FEa- ture and Spatial relaTional regulArization (FESTA), to enable a semantic segmentation CNN to learn discriminative features, while leveraging the unlabelled image pixels. An assumption shared by many unsupervised learning algorithms [229] is that nearby entities often belong to the same class. Based on this assumption, a recent work [230] achieves success in representation learning by encoding neighborhood-relations in the feature space. Inspired by this work, we propose to encode and regularize relations between pixels in both feature and spatial domain, as shown in Fig. 5.20, so that the learned features become more useful for semantic segmentation.

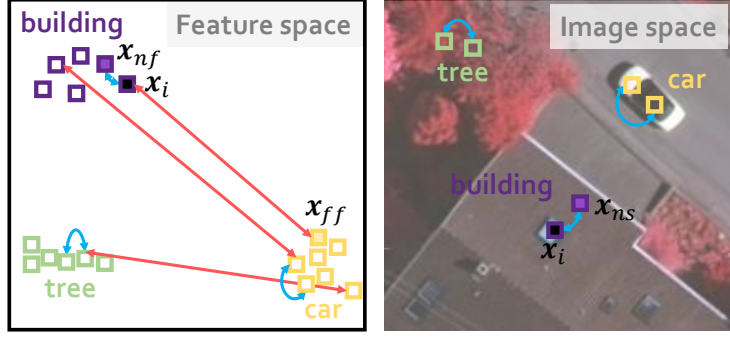


Figure 5.20: Illustration of the proposed FESTA. A Sample x_i belonging to *building* (filled with black) is taken as an example.

Specifically, given a sample x_i (*i.e.*, a CNN feature vector extracted from location i in an image), we first encode its relations to all other samples by measuring the distance in space and feature similarity with respect to all other features in the image. The sample with the smallest similarity is considered as the far-away sample in the feature space, x_{iff} , while that with the highest similarity is defined as the neighboring sample in feature space, x_{inf} . According to the aforementioned proximity assumption, it is highly probable that x_i and x_{inf} belong to the same class, and thus, the distance between them should be as small as possible. In order to prevent a trivial solution in which all features collapse to the same point, x_i and x_{iff} are encouraged to further increase their dissimilarity. We apply a similar reasoning in the spatial domain, since images are smooth in spatial terms. Thus, we take the 8 spatial neighbors of x_i into consideration and chose the one most similar in feature space as the spatial neighbor, x_{ins} . This operation is intended to prevent pairing x_i with a spatial neighbor that belongs to the object boundary.

These priors can be incorporated into the learning objectives by using the following loss function:

$$\begin{aligned} \mathcal{L}_{FESTA} = & \alpha \sum_{i=1}^N \mathcal{D}(x_i, x_{inf}) + \beta \sum_{i=1}^N \mathcal{D}(x_i, x_{ins}) \\ & + \gamma \sum_{i=1}^N \mathcal{S}(x_i, x_{iff}), \end{aligned} \quad (5.20)$$

where \mathcal{D} denotes the euclidean distance and \mathcal{S} represents cosine similarity. α , β , and γ are trade-off parameters representing the significances of the respective terms, and N represents the number of pixels in a given image. By minimizing \mathcal{L}_{FESTA} , x_{inf} and x_{ins} are forced to move closer to x_i , while x_{iff} is pushed far from x_i . In order to jointly exploit the sparse scribbled annotations and FESTA for the network training, the final loss is defined as:

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda \mathcal{L}_{FESTA}, \quad (5.21)$$

where \mathcal{L}_{ce} indicates the categorical cross-entropy loss calculated from pixels with annotations. Furthermore, the predictions of networks trained on scribbled annotations are refined by a fully connected CRF.

Table 5.5: Numerical results on the Vaihingen dataset (%): We show the per-class F_1 score, mean F_1 score, and overall accuracy on the test set. Mean is calculated from results on sparse annotations produced by 4 annotators. Results on dense annotations are provided as reference.

Scribble	Model	Imp. surf.	Build.	Low veg.	Tree	Car	mean F_1	OA
Point	FCN-WL [231]	69.81	75.02	60.25	76.17	12.29	58.71	67.11
	FCN+dCRF [197]	75.37	81.37	61.93	78.50	17.51	62.94	72.53
	FCN-FESTA	74.65	78.64	60.24	76.15	23.65	62.66	71.43
	FCN-FESTA+dCRF	77.62	80.08	60.78	76.70	31.40	65.32	73.65
Line	FCN-WL [231]	78.44	83.45	64.02	79.32	29.01	66.85	76.12
	FCN+dCRF [197]	81.32	84.88	63.71	79.88	38.95	69.75	78.03
	FCN-FESTA	78.12	83.76	65.78	80.49	38.24	69.28	77.24
	FCN-FESTA+dCRF	80.06	84.47	64.35	80.32	43.72	70.58	77.99
Polygon	FCN-WL [231]	76.71	80.03	59.40	78.50	26.28	64.19	74.18
	FCN+dCRF [197]	78.37	80.85	57.92	78.67	29.13	64.99	75.15
	FCN-FESTA	78.98	83.10	62.59	79.91	33.04	67.52	76.65
	FCN-FESTA+dCRF	80.62	83.62	60.79	79.81	40.27	69.02	77.32
Dense	FCN [26]	88.67	92.83	76.32	74.21	86.67	83.74	86.51

5.5.3 Results

We compare a FCN [26] learned using the proposed FESTA (FCN-FESTA) against an FCN learned with weighted loss function (FCN-WL) [231] on sparse annotations. We also report segmentation results of the baseline FCN trained on dense labels. In addition, we study the influence of the fully connected CRF by comparing FCN-FESTA+dCRF and FCN+dCRF [197]. Each model is trained and validated on sparse annotations independently. Per-class F_1 scores, mean F_1 scores, and overall accuracy (OA) are calculated on test images with dense annotations. Considering that each model is learned on labels from four annotators, respectively, we average metrics obtained by each annotator.

Table 5.5 exhibits numerical results on the Vaihingen dataset. FCN-FESTA+dCRF achieves the highest mean F_1 scores in training with all kinds of scribbled annotations, which demonstrates its effectiveness. To be more specific, with point- and polygon-level supervision, FCN-FESTA improves the mean F_1 score by 3.95% and 3.33% compared to FCN-WL, respectively. By refining predictions with dense CRF, FCN-FESTA+dCRF achieves improvements of 2.38% and 4.03% in comparison with FCN+dCRF. It is interesting to observe that line-level scribbles improve the segmentation performance the most, and FCN-FESTA+dCRF learned with such annotations obtains the highest mean F_1 score, 70.58%. Moreover, we note that FESTA can enhance the network capability of recognizing small objects, i.e., *car*, in high resolution aerial images.

The proposed framework is also validated on the Zurich Summer dataset. For more experimental results and analysis, please refer to **Appendix E**.

6 Conclusion and Outlook

6.1 Summary

This dissertation explores deep learning for aerial scene understanding in high resolution remote sensing imagery from the lab to the wild. In this dissertation, we decompose this topic into three specific scene understanding tasks, i.e., aerial scene recognition, multi-label object classification, and semantic segmentation of aerial imagery, and comprehensively analyze their differences in terms of experimental prerequisites. Aiming to present a comprehensive view of aerial scene understanding using deep learning, a thorough literature review is conducted for each task in both laboratory and real-world experimental circumstances. As to the contributions of this dissertation, we reach the four research objectives proposed in Chapter 1 with the following works that are published in five peer-reviewed journals. Specifically,

- **to understand aerial scenes from a fine-grained object perspective**, we propose two multi-label object classification networks, i.e., CA-Conv-BiLSTM and AL-RN-CNN, which encodes label correlations for the final prediction. The former is composed of three indispensable elements: a feature extraction module, a class attention learning layer, and a bidirectional LSTM-based sub-network, and its fundamental component is the bidirectional LSTM-based sub-network that can model underlying class dependency in both directions. The latter comprises a label-wise feature parcel learning module, an attentional region extraction module, and a label relational inference module. To be more specific, the label-wise feature parcel learning module is designed to learn high-level feature parcels, and the attentional region extraction module further generates finer attentional feature parcels by preserving only features located in discriminative regions. Afterward, the label relational inference module reasons about pairwise relations among all labels and exploit these relations for the final prediction. Besides, two multi-label aerial image datasets are proposed for training and evaluating multi-label object classification networks, and experiments on various datasets demonstrate the effectiveness of our networks.
- **to bridge gaps between aerial scene recognition in the lab and wild**, a novel multi-scene recognition network, namely PM-Net, is proposed to tackle both the problem of aerial scene classification in the wild and scarce training samples. To be more specific, PM-Net consists of three key elements: a prototype learning module for encoding prototype representations of variant aerial scenes, a prototype-inhabiting external memory for storing high-level scene prototypes, and a multi-head attention-based memory retrieval module for retrieving associated scene prototypes from the external memory for recognizing multiple scenes in a query aerial image. With this design, PM-Net can learn to predict aerial scenes in the wild by using only a small number of labeled multi-scene images and a huge amount of existing, annotated

single-scene data. Both visual and quantitative results show its performance in aerial scene recognition in the wild.

- **to tackle the problem of data efficiency in multi-scene recognition**, we propose a large-scale dataset, MultiScene, for multi-scene recognition in single images, which is featured by unconstrained multi-scene aerial images and the available both crowdsourced and clean labels. The proposed dataset allows studies in not only recognizing aerial scenes in the wild but also learning from noisy crowdsourced labels. We comprehensively evaluate popular baseline models on both MultiScene-Clean (a subset consisting of only cleanly labeled images) and MultiScene datasets. Experimental results on the former demonstrate that unconstrained multi-scene recognition is still a challenging task, and those on the latter showcase the great potential of exploiting a large number of crowdsourced annotations.
- **to facilitate learning aerial scene parsing models with sparse annotations**, we propose a framework for semantic segmentation of aerial imagery using sparse annotations. In this framework, annotators only need to label several pixels with easy-to-draw scribbles. To exploit these sparse scribbled annotations, we propose the FFeature and Spatial relational regularization (FESTA) method to complement the supervised task with an unsupervised learning signal that accounts for neighborhood structures both in spatial and feature terms.

6.2 Outlook

Deep learning for aerial scene understanding in the laboratory circumstance has been obtaining outstanding achievements during the past few years. In aerial scene recognition, we can see that deep neural networks researches can already reach nearly 100% classification accuracy on some small-scale datasets [34, 46], and over 80% on datasets comprising hundreds of thousands of aerial images [180]. However, there are very few efforts deployed in the field of aerial scene recognition in the wild, where images are unconstrained and more in line with the real-world scenario. Furthermore, multi-scene aerial image datasets required for network training are significantly arduous to produce, which hinders the development of deep learning for multi-scene recognition. Hence looking into the future, large-scale datasets, where images are captured with no constraints and assigned multiple scene labels, are in an urgent need for the remote sensing community. Besides, to alleviate such annotation burden, how to jointly leverage a large number of single-scene aerial image datasets and limited annotated multi-scene images is also a promising research direction. Specifically, although deep learning models are capable of automatically extracting discriminative features after trained on massive annotated images, they might suffer from poor generalization ability due to their enormous learnable parameters. Hence, feature learning strategies that are learned from single-scene images may not perform as expected on multi-scene images, which imposes significant challenges to learning domain-invariant scene representations. Another direction is to train networks to identify multiple objects and scenes simultaneously, as objects are fundamental components of scenes and invariant to the completeness and distributions of scenes in an image. To do so, graph networks and attention mechanisms showcase the great potential in modeling underlying relationships for enhancing the network performance of recognizing aerial scenes in unconstrained aerial images.

Semantic segmentation of aerial imagery is a longstanding problem and have attracted numerous researches that are conducted under the laboratory circumstance. Albeit considerable progress has been made, their performance can not be made the most of in the real-world scenario owing to data insufficiency. This is even stern in applications that are faced with unseen targets and require fast responses. Opportunities always arise from challenges. The problem of insufficient training samples in the wild can be addressed from the following perspectives:

- Enlarging pixel-wise annotated aerial image datasets by increasing the number of high resolution aerial images and taking thematic classes into consideration. As can be observed in existing datasets, their predefined categories are often based on land cover and land use types and provide a primary object-level understanding of aerial images. Thus, networks trained on them can only interpret *what they see on the earth* but are not able to *reason about thematic meanings*. For example, networks trained on ISPRS datasets can identify cars and impervious surfaces but fail to perceive which pixel belongs to parking lots or roads that are essential in urban management and traffic monitoring. Besides, expanding dataset scales is beneficial for alleviating the overfitting problem in network training and improving the generalization capability.
- Reasoning about pixel relations for constructing more discriminative embedding spaces so that unknown pixels can be interpreted by propagating labels of a few pixels. Recently, Transformer and its variants are popular in visual recognition tasks due to its revolutionary performance of learning long-range relationships between entities and allowing strong information flow among them. Such capabilities can as well benefit semantic segmentation with sparse and incomplete annotations by learning a more compact and discriminative feature space. Nonetheless, Transformer-based models are data hunger, which imposes a gargantuan challenge to inventing parameter-efficient alternatives of self-attention mechanism. To tackle the problem of limited training samples, unsupervised self-learning is introduced to pre-train networks on large-scale upstream datasets before fine-tuning them on downstream tasks.
- Learning more general feature representations from large-scale cross-domain and cross-task datasets. For instance, even a car shows variant appearances in aerial and natural images due to their different viewing perspectives (i.e., the nadir and side view), humans can readily recognize them with high confidence. This is attributed to the inherent capability of learning general and discriminative features by continuously observing the world. Recalling that the number of remote sensing image datasets is huge, and these datasets are painless to access, continual learning can showcase its potential in learning intrinsic feature representations for preceding tasks having few labels. Moreover, in the computer vision community, there are large volumes of high-quality pixel-wise annotations for visual recognition tasks, which may further boost the learning efficiency.

The training of deep learning-based algorithms is heavily dependent on access to high-quality annotated image data. The more and higher-quality annotations are available, the better-performed networks can be learned. However, as life is always hard, either pixel-wise or image-level annotations often suffer from deficiency, noise, and incompleteness.

6 Conclusion and Outlook

With respect to the three problems, learning with few shots, noisy labels, and partial labels is exceedingly challenging but practical in the real-world scenario. By integrating prior knowledge, such as properties and underlying correlations, and semi-supervised learning schemes, deep learning algorithms are expected to perform well in more practical applications.

Towards general Artificial Intelligence (AI) for aerial scene understanding, incorporating natural languages is necessary, as they contain rich semantic information. Researches [232, 233] about word embedding demonstrate that words having relevant semantics or frequently co-occurring are closer to each other in the word embedding space. This phenomenon can also be observed in visual tasks that images assigned common scene or object labels show high similarities in the feature space. Therefore, it is worthwhile to exploit visual and linguistic cues for aerial scene understanding, as they deliver homogeneous information of the real world but in a complementary way.

List of Figures

1.1	Examples of different scene understanding tasks. Given an example aerial image (a), (b) scene recognition aims at predicting the scene category, while (c) multi-label object classification targets at identifying multiple co-existing objects. In (d) semantic segmentation, the category of every pixel should be inferred. The legendary of (d) can be referred to in (c).	2
2.1	The architecture of LeNet-5 [4]. Each plane represents a feature map. <i>C1</i> , <i>C3</i> , and <i>C5</i> are convolutional layers with 5×5 filters. <i>S2</i> and <i>S4</i> are subsampling layers that halve the width and height of feature maps. <i>F6</i> is a fully-connected layer.	6
2.2	The architecture of VGG-16. <i>Conv</i> and <i>FC</i> stand for convolutional layer and fully-connected layer, respectively.	6
2.3	Architectures of (a) Inception-v1 and (b) Inception-v3 modules [11].	7
2.4	The architecture of ResNet-34 [15].	8
2.5	Architectures of (a) ResNet and (b) ResNeXt blocks of equivalent complexities [16]. The configuration of each layer is denoted as <i>the channel dimension of inputs, the size of convolutional filters, and the channel dimension of outputs</i>	8
2.6	The illustration of three consecutive dense blocks [17]. In each block, darker nodes denote higher-level feature maps, while light nodes represent low-level features. Each node is connected to all its subsequent nodes in the common block. Curved arrows denote identity mappings.	9
2.7	Illustration of (a) pointwise and (b) depthwise convolutions. Planes of variant colors indicate different feature channels. Given feature maps with size of $H \times W \times C$ (representing the height, width, and channel), the size of each pointwise convolutional filter is $1 \times 1 \times C$, and that of a depthwise filter is $K \times K \times 1$. Notably, the number of depthwise convolutional filters is required to set as C , and K is arbitrarily defined.	9
2.8	Illustration of (a) FCN, (b) SegNet, and (c) U-Net. Gray bars indicate convolutional and pooling layers in the encoder, while orange bars represent deconvolutional and convolutional layers in the decoder. Besides, Green bars denote upsampling and convolutional layers in SegNet. Arrows represent skip connections.	10
3.1	Caption for LOF	12
3.2	Visualization of attentional regions captured by VGG-16 in recognizing (a) <i>intersection</i> and (d) <i>beach</i> . (b) and (e) are CAMs generated by shallow convolutional layers of VGG-16, while (c) and (f) are CAMs extracted by deep layers.	14

3.3	Illustration of (a) multi-scale feature extraction, (b) multi-level feature extraction, and (c) rotation-equivariant feature extraction. Notably, in (a), pooling/convolution is conducted in sliding windows, and both images and feature maps can be taken as input.	15
3.4	Illustration of (a) channel and (b) spatial attention modules, and (c) self-attention mechanism. In (a), global average/max pooling is first conducted, and output feature vectors are fed to fully-connected layers for learning channel attentions. In (b), a spatial attention mask is learned with 1×1 convolutions. To apply (c), feature maps are reshaped along the spatial dimension, which yields a sequence of feature vectors, before they are linearly transformed to query, key, and value.	17
3.5	Illustration of (a) binary relevance and (b) label relation mining approaches.	19
3.6	Example high resolution aerial images delineating (a) <i>industrial</i> , (b) <i>residential</i> , and (c) <i>parking lot</i> but sharing common object labels, <i>car</i> and <i>pavement</i>	20
3.7	Example aerial images from variant datasets (from left to right: AID, UCM, RSI-CB256, and RSD46-WHU) with respect to different scenes (From top to bottom: <i>airport</i> , <i>harbor</i> , <i>parking lot</i> , <i>river</i> , <i>residential</i> , and <i>storage tanks</i>). Data source platforms are denoted in the bottom row.	24
3.8	Example aerial images with multiple labels from DFC15-mul ((a) and (d)), BigEarthNet ((b) and (c)), and MLRSNet ((e) and (f)) datasets. Their labels: (a) <i>Impervious</i> , <i>vegetation</i> , <i>building</i> , and <i>car</i> . (b) <i>permanently irrigated land</i> , <i>sclerophyllous vegetation</i> , <i>beaches</i> , <i>dunes</i> , <i>sands</i> , <i>estuaries</i> , <i>sea</i> and <i>ocean</i> . (c) <i>mountain</i> , <i>snow</i> , and <i>snowberg</i> . (d) <i>Water</i> , <i>clutter</i> , and <i>boat</i> . (e) <i>discontinuous urban fabric</i> , <i>non-irrigated arable land</i> , <i>land principally occupied by agriculture</i> , and <i>broad-leaved forest</i> (f) <i>buildings</i> , <i>crosswalk</i> , <i>grass</i> , <i>trees</i> , <i>cars</i> , <i>pavement</i> , <i>road</i> , and <i>intersection</i>	26
4.1	Illustration of querying building features on the OSM platform.	30
4.2	Aerial images taken over urban residential areas in (a) Germany, (b) China, and (c) the US and are provided by Google Earth. Albeit identical functions, they share variant architectures, visual appearances, and layouts. . .	32
4.3	Illustration of sparse point-, scribble-, and polygon-level annotations. . . .	33
4.4	Illustration of two pipelines for learning with sparse segmentation. In (a), pixels are clustered into superpixels for label and semantic propagation before training networks. In (b), predicted masks are fed to a graph model for regularization based on input image contexts.	34
4.5	Example unconstrained aerial images in the MAI dataset. Their scene-level multiple labels are here: (a) farmland and residential; (b) baseball, woodland, parking lot, and tennis court; (c) commercial, parking lot, and residential; (d) woodland, residential, river, and runway; (e) river and storage tanks; (f) beach, woodland, residential, and sea.	36
4.6	Caption for LOF	37
4.7	Example scribbled pixel-wise annotations of the ISPRS Vaihingen dataset. The 2nd and 3rd columns are made by experts, and the left two are created by non-experts. Legend— <i>white</i> : impervious surfaces, <i>blue</i> : buildings, <i>cyan</i> : low vegetation, <i>green</i> : trees, <i>yellow</i> : cars.	38

5.1	The contribution matrix of labels in UCM dataset. Labels at X-axis represent referenced classes C_r , while labels at Y-axis are potential co-occurrence classes C_p . Conditional probabilities $P(C_p C_r)$ of each class pair are present in corresponding blocks.	43
5.2	The architecture of the proposed CA-Conv-BiLSTM for the multi-label classification of aerial images.	43
5.3	Illustration of the bidirectional structure. The direction of the upper stream is opposite to that of the lower stream. Notably, \mathbf{h}'_{l-1} , \mathbf{c}'_{l-1} denotes the activation and memory cell in the upper stream at the time step, which corresponds to class $l - 1$ for convenience (considering that the subsequent time step is usually denoted as $l + 1$).	46
5.4	Example class attention maps of (a) images in UCM multi-label dataset with respect to (b) bare soil, (c) building, (d) car, (e) court, (f) grass, (g) pavement, (h) tree, and (i) water. Red indicates strong activations, while blue represents non-activations. Besides, normalization is performed based on each row for a fair comparison among class attention maps of the same images.	48
5.5	The architecture of the proposed attention-aware label relational reasoning network.	49
5.6	Illustration of the attentional region extraction module. Green dots in the left image indicate the feature parcel grid $G_{\mathbf{X}_l}$. White dots in the middle image represent the attentional feature parcel grid $G_{\mathbf{X}_l^{attn}}$, while those in the right image indicate re-coordinated $G_{\mathbf{X}_l^{attn}}$. Notably, the structure of re-coordinated $G_{\mathbf{X}_l^{attn}}$ is identical to that of $G_{\mathbf{X}_l}$, and values of pixels located at grid points in re-coordinated $G_{\mathbf{X}_l^{attn}}$ are obtained from those in $G_{\mathbf{X}_l^{attn}}$. For example, the pixel at the left top corner grid point in re-coordinated $G_{\mathbf{X}_l^{attn}}$ is assigned with the value of that at the left top corner of $G_{\mathbf{X}_l^{attn}}$	49
5.7	Illustration of the label relation module.	51
5.8	Example attentional regions for car, bare soil (soil), building (build.), pavement (pave.), court, and tank in various scenes (a)-(d) in the UCM multi-label dataset.	52
5.9	Illustration of how humans learn to perceive unconstrained aerial images being composed of multiple scenes. We first learn and memorize individual aerial scenes. Then we can possess the capability of understanding complex scenarios by learning from only a limited number of hard instances. We believe by simulating this learning process, a deep neural network can also learn to interpret multi-scene aerial images.	53
5.10	Architecture of the proposed PM-Net. Particularly, we first learn scene prototypes \mathbf{p}_s from well-annotated single-scene aerial images and then store them in the external memory \mathbf{M} of PM-Net. Afterwards, given a query multi-scene image, a multi-head attention-based memory retrieval module is devised to retrieve scene prototypes that are relevant to the query image, yielding \mathbf{z}' for the prediction of multiple labels. f_ϕ denotes the embedding function, and its output is a D -dimensional feature vector. S and H represent numbers of scenes and heads, respectively. L and U denote channel dimensions of the key and value in the memory retrieval module.	54

5.11	T-SNE visualization of image representations and scene prototypes learned by VGGNet on the UCM dataset. Dots in the same color represent features of images belonging to the same scene, and stars denote scene prototypes. .	57
5.12	Examples of images utilized in (a) single-scene and (b) multi-scene recognition tasks. In (a), each aerial image is assigned one scene label, while in (b), labels of all present scenes are inferred. In comparison with (b), (a) might suffer from partial scene understanding, as only one label is predicted even if there indeed exist multiple scenes in an image. For a clear visualization, locations of scenes are marked in (b).	58
5.13	Example multi-scene aerial images with their crowdsourced and clean annotations in the MultiScene dataset.	59
5.14	Coordinate distributions and examples of multi-scene aerial images in our dataset. Red dots denote images with both crowdsourced and clean labels, and cyan dots represent images with only crowdsourced scene labels. . . .	59
5.15	Sample distributions of all scene categories in our dataset. Each cyan bar indicates the number of images assigned only OSM labels with respect to each scene category, and red bars represent numbers of images with both OSM and clean labels.	60
5.16	The number of images associated with different numbers of scenes. Y-axis indicates the number of scenes, and X-axis represents the number of images. The legend is the same as that in Figure 5.15.	60
5.17	Comparisons of the performance of networks trained on images with clean (light-color bars) and crowdsourced (dark-color bars) annotations, respectively. For each network, the left four bars represent class-based scores, mAPs and CF ₁ , while the right four bars indicate EF ₁ and OF ₁ scores. . .	61
5.18	Comparisons of different levels of scribbled annotations. Trees (marked as green) are taken as an example here. Images from left to right are (a) an aerial image, (b) point-, (c) line- and (d) polygon-level scribbled annotations, and (e) dense pixel-wise labels.	62
5.19	Example polygon-level annotations of an image (ID: 13) on the Vaihingen dataset. Annotations of cars are zoomed in to illustrate that annotations should include variant visual appearances for one class. Legend— white : impervious surfaces, blue : buildings, cyan : low vegetation, green : trees, yellow : cars.	63
5.20	Illustration of the proposed FESTA. A Sample x_i belonging to <i>building</i> (filled with black) is taken as an example.	64

List of Tables

3.1	An overview of existing public single-scene aerial image datasets.	23
3.2	An overview of existing public multi-label aerial image datasets.	23
3.3	An overview of existing high resolution image semantic segmentation datasets.	25
4.1	The total numbers of pixels labeled with sparse point-, line-, and polygon-level annotations (middle three columns) and dense annotations (right column) in the Vaihingen and Zurich Summer datasets.	38
5.1	Quantitative Results on UCM Multi-label Dataset (%)	47
5.2	Comparisons of the classification performance on UCM Multi-label Dataset (%).	52
5.3	Numerical Results on UCM2MAI (%).	56
5.4	Numerical results of baseline models on the MultiScene-Clean dataset (%). Models are trained and tested on cleanly-labeled images, and the best scores are shown in bold.	61
5.5	Numerical results on the Vaihingen dataset (%): We show the per-class F_1 score, mean F_1 score, and overall accuracy on the test set. Mean is calculated from results on sparse annotations produced by 4 annotators. Results on dense annotations are provided as reference.	65

Bibliography

- [1] B. Manjunath and W. Ma. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):837–842, 1996.
- [2] D. Loew. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [3] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.
- [4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [5] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1097–1105, 2012.
- [6] X. X. Zhu, D. Tuia, L. Mou, G. Xia, L. Zhang, F. Xu, and F. Fraundorfer. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine*, 5(4):8–36, 2017.
- [7] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. Johnson. Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 152:166–177, 2019.
- [8] G. Cheng, X. Xie, J. Han, L. Guo, and G. Xia. Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:3735–3756, 2020.
- [9] J. Sánchez and F. Perronnin. High-dimensional signature compression for large-scale image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1665–1672, 2011.
- [10] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014.
- [11] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the Inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, 2016.
- [12] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015.
- [13] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, pages 448–456, 2015.
- [14] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi. Inception-v4, Inception-ResNet and the impact of residual connections on learning. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2017.
- [15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

Bibliography

- [16] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5987–5995, 2017.
- [17] G. Huang, Z. Liu, L. V. D. Maaten, and K. Weinberger. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4700–4708, 2017.
- [18] A. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv:1704.04861*, 2017.
- [19] F. Chollet. Xception: Deep learning with depthwise separable convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1251–1258, 2017.
- [20] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen. MobileNetV2: Inverted residuals and linear bottlenecks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520, 2018.
- [21] X. Zhang, X. Zhou, M. Lin, and J. Sun. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6848–6856, 2018.
- [22] F. Iandola, S. Han, M. Moskewicz, K. Ashraf, W. Dally, and K. Keutzer. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and 0.5 MB model size. In *International Conference on Learning Representations (ICLR)*, 2017.
- [23] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, and Q. Le. MnasNet: Platform-aware neural architecture search for mobile. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [24] B. Zoph and Q. Le. Neural architecture search with reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2017.
- [25] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and F. Li. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009.
- [26] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015.
- [27] V. Badrinarayanan, A. Kendall, and R. Cipolla. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017.
- [28] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pages 234–241, 2015.
- [29] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2017.
- [30] L. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv:1706.05587*, 2017.
- [31] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2881–2890, 2017.
- [32] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 886–893, 2005.

- [33] M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.
- [34] Y. Yang and S. Newsam. Bag-of-visual-words and spatial extensions for land-use classification. In *ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*, pages 270–279, 2010.
- [35] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *European Conference on Computer Vision (ECCV)*, pages 143–156, 2010.
- [36] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.
- [37] A. Bosch, A. Zisserman, and X. Munoz. Scene classification via pLSA. In *European Conference on Computer Vision (ECCV)*, pages 517–530, 2006.
- [38] Y. Zhong, Q. Zhu, and L. Zhang. Scene classification based on the multifeature fusion probabilistic topic model for high spatial resolution remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 53(11):6207–6222, 2015.
- [39] R. Kusumaningrum, H. Wei, R. Manurung, and A. Murni. Integrated visual vocabulary in latent Dirichlet allocation-based scene classification for IKONOS image. *Journal of Applied Remote Sensing*, 8(1):083690, 2014.
- [40] J. dos Santos, O. Penatti, and R. da Silva Torres. Evaluating the potential of texture and color descriptors for remote sensing image retrieval and classification. In *International Conference on Computer Vision Theory and Applications (VISAPP)*, pages 203–208, 2010.
- [41] L. Chen, W. Yang, K. Xu, and T. Xu. Evaluation of local features for scene classification using vhr satellite images. In *Joint Urban Remote Sensing Event (JURSE)*, pages 385–388, 2011.
- [42] V. Risojević and Z. Babić. Fusion of global and local descriptors for remote sensing image classification. *IEEE Geoscience and Remote Sensing Letters*, 10(4):836–840, 2012.
- [43] G. Cheng, L. Guo, T. Zhao, J. Han, H. Li, and J. Fang. Automatic landslide detection from remote-sensing imagery using a scene classification method based on BoVW and pLSA. *International Journal of Remote Sensing*, 34(1):45–59, 2013.
- [44] Q. Zhu, Y. Zhong, B. Zhao, G. Xia, and L. Zhang. Bag-of-visual-words scene classifier with local and global features for high spatial resolution remote sensing imagery. *IEEE Geoscience and Remote Sensing Letters*, 13(6):747–751, 2016.
- [45] J. Hu, T. Jiang, X. Tong, G. Xia, and L. Zhang. A benchmark for scene classification of high spatial resolution remote sensing imagery. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 5003–5006, 2015.
- [46] G. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, and X. Lu. AID: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(7):3965–3981, 2017.
- [47] . Long, G. Xia, S. Li, W. Yang, M. Yang, X. X. Zhu, L. Zhang, and D. Li. On creating benchmark dataset for aerial image interpretation: Reviews, guidances and million-aid. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:4205–4230, 2021.
- [48] Y. Hua, L. Mou, and X. X. Zhu. LAHNet: A convolutional neural network fusing low- and high-level features for aerial scene classification. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 4728–4731, 2018.
- [49] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2929, 2016.

- [50] D. Marmanis, M. Datcu, T. Esch, and U. Stilla. Deep learning earth observation classification using ImageNet pretrained networks. *IEEE Geoscience and Remote Sensing Letters*, 13(1):105–109, 2015.
- [51] S. Chaib, H. Liu, Y. Gu, and H. Yao. Deep feature fusion for VHR remote sensing scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(8):4775–4784, 2017.
- [52] E. Li, J. Xia, P. Du, C. Lin, and A. Samat. Integrating multilayer features of convolutional neural networks for remote sensing scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(10):5653–5665, 2017.
- [53] M. Castelluccio, G. Poggi, C. Sansone, and L. Verdoliva. Land use classification in remote sensing images by convolutional neural networks. *arXiv:1508.00092*, 2015.
- [54] Y. Yuan, J. Fang, X. Lu, and Y. Feng. Remote sensing image scene classification using rearranged local features. *IEEE Transactions on Geoscience and Remote Sensing*, 57(3):1779–1792, 2018.
- [55] F. Hu, G. Xia, J. Hu, and L. Zhang. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing*, 7(11):14680–14707, 2015.
- [56] Y. Liu, Y. Zhong, and Q. Qin. Scene classification based on multiscale convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing*, 56(12):7109–7121, 2018.
- [57] J. Xie, N. He, L. Fang, and A. Plaza. Scale-free convolutional neural network for remote sensing scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(9):6916–6928, 2019.
- [58] R. Tombe and S. Viriri. Adaptive deep co-occurrence feature learning based on classifier-fusion for remote sensing scene classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:155–164, 2020.
- [59] T. Tian, L. Li, W. Chen, and H. Zhou. SEMSDNet: A multi-scale dense network with attention for remote sensing scene classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2021.
- [60] L. Li, T. Tian, H. Li, and L. Wang. SE-HRNet: A deep high-resolution network with attention for remote sensing scene classification. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 533–536, 2020.
- [61] G. Zhang, W. Xu, W. Zhao, C. Huang, E. Yk, Y. Chen, and J. Su. A multi-scale attention (msa) network for remote sensing scene images classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2021.
- [62] J. Kang and B. Demir. Band-wise multi-scale cnn architecture for remote sensing image scene classification. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1687–1690, 2020.
- [63] X. Lu, H. Sun, and X. Zheng. A feature aggregation convolutional neural network for remote sensing scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(10):7894–7906, 2019.
- [64] X. Zhang, W. An, J. Sun, H. Wu, W. Zhang, and Y. Du. Best representation branch model for remote sensing image scene classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:9768–9780, 2021.
- [65] H. Wan, J. Chen, Z. Huang, Y. Feng, Z. Zhou, X. Liu, B. Yao, and T. Xu. Lightweight channel attention and multiscale feature fusion discrimination for remote sensing scene classification. *IEEE Access*, 9:94586–94600, 2021.

- [66] S. Mei, K. Yan, M. Ma, X. Chen, S. Zhang, and Q. Du. Remote sensing scene classification using sparse representation-based framework with deep feature fusion. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:5867–5878, 2021.
- [67] X. Hu, P. Zhang, and Q. Zhang. A novel framework of CNN integrated with Adaboost for remote sensing scene classification. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 2643–2646, 2020.
- [68] J. Shen, T. Zhang, Y. Wang, R. Wang, Q. Wang, and M. Qi. A dual-model architecture with grouping-attention-fusion for remote sensing scene classification. *Remote Sensing*, 13(3):433, 2021.
- [69] X. Sun, Q. Zhu, and Q. Qin. A multi-level convolution pyramid semantic fusion framework for high-resolution remote sensing image scene classification and annotation. *IEEE Access*, 9:18195–18208, 2021.
- [70] X. Tang, Q. Ma, X. Zhang, F. Liu, J. Ma, and L. Jiao. Attention consistent network for remote sensing scene classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:2030–2045, 2021.
- [71] J. Ma, Q. Ma, X. Tang, X. Zhang, C. Zhu, Q. Peng, and L. Jiao. Remote sensing scene classification based on global and local consistent network. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 537–540, 2020.
- [72] Z. Zeng, X. Chen, and Z. Song. MGFN: A multi-granularity fusion convolutional neural network for remote sensing scene classification. *IEEE Access*, 9:76038–76046, 2021.
- [73] K. Qi, C. Yang, C. Hu, Y. Shen, S. Shen, and H. Wu. Rotation invariance regularization for remote sensing image scene classification with convolutional neural networks. *Remote Sensing*, 13(4):569, 2021.
- [74] H. Xie, Y. Chen, and P. Ghamisi. Remote sensing image scene classification via label augmentation and intra-class constraint. *Remote Sensing*, 13(13):2566, 2021.
- [75] D. Guo, Y. Xia, and X. Luo. Self-supervised GANs with similarity loss for remote sensing image scene classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:2508–2521, 2021.
- [76] Z. Liu and L. Ma. Class-wise adversarial transfer network for remote sensing scene classification. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1357–1360, 2020.
- [77] Y. Li, Y. Zhang, and Z. Zhu. Error-tolerant deep learning for remote sensing image scene classification. *IEEE Transactions on Cybernetics*, 51(4):1756–1768, 2020.
- [78] Y. Liu, C. Suen, Y. Liu, and L. Ding. Scene classification using hierarchical Wasserstein cnn. *IEEE Transactions on Geoscience and Remote Sensing*, 57(5):2494–2509, 2018.
- [79] Y. Li, R. Chen, Y. Zhang, M. Zhang, and L. Chen. Multi-label remote sensing image scene classification by combining a convolutional neural network and a graph neural network. *Remote Sensing*, 12(23):4003, 2020.
- [80] J. Luo, Y. Wang, Y. Ou, B. He, and B. Li. Neighbor-based label distribution learning to model label ambiguity for aerial scene classification. *Remote Sensing*, 13(4):755, 2021.
- [81] Q. Zhu, X. Fan, Y. Zhong, Q. Guan, L. Zhang, and D. Li. Super resolution generative adversarial network based image augmentation for scene classification of remote sensing images. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 573–576, 2020.
- [82] L. Chen, Z. Xu, Q. Li, J. Peng, S. Wang, and H. Li. An empirical study of adversarial examples on remote sensing image scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(9), 2021.

Bibliography

- [83] Y. Xu, B. Du, and L. Zhang. Assessing the threat of adversarial examples on deep neural networks for remote sensing scene classification: Attacks and defenses. *IEEE Transactions on Geoscience and Remote Sensing*, 59(2):1604–1617, 2020.
- [84] R. Minetto, M. Segundo, and S. Sarkar. Hydra: An ensemble of convolutional neural networks for geospatial land classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(9):6530–6541, 2019.
- [85] F. Zhang, B. Du, and L. Zhang. Scene classification via a gradient boosting random convolutional network framework. *IEEE Transactions on Geoscience and Remote Sensing*, 54(3):1793–1802, 2015.
- [86] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han. When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs. *IEEE Transactions on Geoscience and Remote Sensing*, 56(5):2811–2821, 2018.
- [87] C. Xu, G. Zhu, and J. Shu. Robust joint representation of intrinsic mean and kernel function of Lie group for remote sensing scene classification. *IEEE Geoscience and Remote Sensing Letters*, 18(5):796–800, 2020.
- [88] H. Li, Z. Cui, Z. Zhu, L. Chen, J. Zhu, H. Huang, and C. Tao. Rs-metanet: Deep metametric learning for few-shot remote sensing scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(8):6983–6994, 2021.
- [89] Z. Yuan, W. Huang, L. Li, and X. Luo. Few-shot scene classification with multi-attention deepemd network in remote sensing. *IEEE Access*, 9:19891–19901, 2020.
- [90] X. Li, F. Pu, R. Yang, R. Gui, and X. Xu. AMN: Attention metric network for one-shot remote sensing image scene classification. *Remote Sensing*, 12(24):4046, 2020.
- [91] Q. Zeng, J. Geng, K. Huang, W. Jiang, and J. Guo. Prototype calibration with feature generation for few-shot remote sensing image scene classification. *Remote Sensing*, 13(14):2728, 2021.
- [92] Y. Li, D. Kong, Y. Zhang, Y. Tan, and L. Chen. Robust deep alignment network with remote sensing knowledge graph for zero-shot and generalized zero-shot remote sensing image scene classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 179:145–158, 2021.
- [93] Y. Li, Z. Shao, X. Huang, B. Cai, and S. Peng. Meta-FSEO: A meta-learning fast adaptation with self-supervised embedding optimization for few-shot remote sensing scene classification. *Remote Sensing*, 13(14):2776, 2021.
- [94] P. Zhang, Y. Bai, D. Wang, B. Bai, and Y. Li. Few-shot classification of aerial scene images via meta-learning. *Remote Sensing*, 13(1):108, 2021.
- [95] L. Li, J. Han, X. Yao, G. Cheng, and L. Guo. DLA-Matchnet for few-shot remote sensing image scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(9):7844–7853, 2021.
- [96] T. Lasloun, H. Alhichri, Y. Bazi, and N. Alajlan. SSDAN: Multi-source semi-supervised domain adaptation network for remote sensing scene classification. *Remote Sensing*, 13(19):3861, 2021.
- [97] C. Ma, D. Sha, and X. Mu. Unsupervised adversarial domain adaptation with error-correcting boundaries and feature adaption metric for remote-sensing scene classification. *Remote Sensing*, 13(7):1270, 2021.
- [98] J. Zhang, J. Liu, B. Pan, and Z. Shi. Domain adaptation based on correlation subspace dynamic distribution alignment for remote sensing image scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 58(11):7920–7930, 2020.

- [99] J. Zhang, J. Liu, L. Shi, B. Pan, and X. Xu. An open set domain adaptation network based on adversarial learning for remote sensing image scene classification. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1365–1368, 2020.
- [100] J. Xia, Y. Ding, and L. Tan. Urban remote sensing scene recognition based on lightweight convolution neural network. *IEEE Access*, 9:26377–26387, 2021.
- [101] A. Byju, G. Sumbul, B. Demir, and L. Bruzzone. Remote-sensing image scene classification with deep neural networks in JPEG 2000 compressed domain. *IEEE Transactions on Geoscience and Remote Sensing*, 59(4):3458–3472, 2020.
- [102] C. Peng, Y. Li, L. Jiao, and R. Shang. Efficient convolutional neural architecture search for remote sensing image scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 2020.
- [103] Y. Wan, Y. Zhong, A. Ma, J. Wang, and R. Feng. RSSM-Net: Remote sensing image scene classification based on multi-objective neural architecture search. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1369–1372, 2020.
- [104] Y. Chen, W. Teng, Z. Li, Q. Zhu, and Q. Guan. Cross-domain scene classification based on a spatial generalized neural architecture search for high spatial resolution remote sensing images. *Remote Sensing*, 13(17):3460, 2021.
- [105] A. Ma, Y. Wan, Y. Zhong, J. Wang, and L. Zhang. SceneNet: Remote sensing scene classification deep learning network using multi-objective neural evolution architecture search. *ISPRS Journal of Photogrammetry and Remote Sensing*, 172:171–188, 2021.
- [106] N. He, L. Fang, S. Li, A. Plaza, and J. Plaza. Remote sensing scene classification using multilayer stacked covariance pooling. *IEEE Transactions on Geoscience and Remote Sensing*, 56(12):6899–6910, 2018.
- [107] L. Dan and X. Li. Relationships excavating of augmented feature for remote sensing scene classification. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1361–1364, 2020.
- [108] S. Woo, J. Park, J. Lee, and I. Kweon. CBAM: Convolutional block attention module. In *European Conference on Computer Vision (ECCV)*, pages 3–19, 2018.
- [109] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5998–6008, 2017.
- [110] J. Kong, Y. Gao, Y. Zhang, H. Lei, Y. Wang, and H. Zhang. Improved attention mechanism and residual network for remote sensing image scene classification. *IEEE Access*, 2021.
- [111] C. Shi, X. Zhao, and L. Wang. A multi-branch feature fusion strategy based on an attention mechanism for remote sensing image scene classification. *Remote Sensing*, 13(10):1950, 2021.
- [112] J. Kim and M. Chi. Saffnet: Self-attention-based feature fusion network for remote sensing few-shot scene classification. *Remote Sensing*, 13(13):2532, 2021.
- [113] M. Li, L. Lei, Y. Tang, Y. Sun, and G. Kuang. An attention-guided multilayer feature aggregation network for remote sensing image scene classification. *Remote Sensing*, 13(16):3113, 2021.
- [114] R. Cao, L. Fang, T. Lu, and N. He. Self-attention-based deep feature fusion for remote sensing scene classification. *IEEE Geoscience and Remote Sensing Letters*, 18(1):43–47, 2020.
- [115] Q. Bi, K. Qin, H. Zhang, and G. Xia. Local semantic enhanced convnet for aerial scene recognition. *IEEE Transactions on Image Processing*, 30:6498–6511, 2021.
- [116] L. Fu, D. Zhang, and Q. Ye. Recurrent thrifty attention network for remote sensing scene recognition. *IEEE Transactions on Geoscience and Remote Sensing*, 2020.

Bibliography

- [117] K. Qi, C. Yang, C. Hu, Y. Shen, and H. Wu. Deep object-centric pooling in convolutional neural network for remote sensing scene classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:7857–7868, 2021.
- [118] C. Ma, X. Mu, R. Lin, and S. Wang. Multilayer feature fusion with weight adjustment based on a convolutional neural network for remote sensing scene classification. *IEEE Geoscience and Remote Sensing Letters*, 18(2):241–245, 2020.
- [119] J. Shen, C. Zhang, Y. Zheng, and R. Wang. Decision-level fusion with a pluginable importance factor generator for remote sensing image scene classification. *Remote Sensing*, 13(18):3579, 2021.
- [120] H. Sun, S. Li, X. Zheng, and X. Lu. Remote sensing scene classification by gated bidirectional network. *IEEE Transactions on Geoscience and Remote Sensing*, 58(1):82–96, 2019.
- [121] Q. Wang, S. Liu, J. Chanussot, and X. Li. Scene classification with recurrent attention of vhr remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 57(2):1155–1167, 2018.
- [122] S. Tong, K. Qi, Q. Guan, Q. Zhu, C. Yang, and J. Zheng. Remote sensing scene classification using spatial transformer fusion network. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 549–552, 2020.
- [123] C. Wang, Y. Wu, Y. Wang, Y. Chen, and Y. Gao. Multilevel capsule weighted aggregation network based on a decoupled dynamic filter for remote sensing scene classification. *IEEE Access*, 9:125309–125319, 2021.
- [124] R. Bhagwat and B. Shankar. A novel multilabel classification of remote sensing images using XGBoost. In *International Conference for Convergence in Technology (I2CT)*, pages 1–5, 2019.
- [125] A. Zeggada and F. Melgani. Multilabeling UAV images with autoencoder networks. In *Joint Urban Remote Sensing Event (JURSE)*, pages 1–4, 2017.
- [126] A. Zeggada, F. Melgani, and Y. Bazi. A deep learning approach to UAV image multilabeling. *IEEE Geoscience and Remote Sensing Letters*, 14(5):694–698, 2017.
- [127] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979.
- [128] T. Chen and C. Guestrin. XGBoost: A scalable tree boosting system. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.
- [129] B. Zegeye and B. Demir. A novel active learning technique for multi-label remote sensing image scene classification. In *Image and Signal Processing for Remote Sensing*, volume 10789, page 107890B, 2018.
- [130] L. Bashmal, Y. Bazi, M. Al Rahhal, H. Alhichri, and N. Al Ajlan. UAV image multi-labeling with data-efficient transformers. *Applied Sciences*, 11(9):3974, 2021.
- [131] A. Zeggada, S. Benbraika, F. Melgani, and Z. Mokhtari. Multilabel conditional random field classification for UAV images. *IEEE Geoscience and Remote Sensing Letters*, 15(3):399–403, 2018.
- [132] B. Chaudhuri, B. Demir, S. Chaudhuri, and L. Bruzzone. Multilabel remote sensing image retrieval using a semisupervised graph-theoretic method. *IEEE Transactions on Geoscience and Remote Sensing*, 56(2):1144–1158, 2017.
- [133] O. Dai, B. Demir, B. Sankur, and L. Bruzzone. A novel system for content-based retrieval of single and multi-label high-dimensional remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(7):2473–2490, 2018.

- [134] P. Zhu, Y. Tan, L. Zhang, Y. Wang, J. Mei, H. Liu, and M. Wu. Deep learning for multilabel remote sensing image annotation with dual-level semantic concepts. *IEEE Transactions on Geoscience and Remote Sensing*, 58(6):4047–4060, 2020.
- [135] Y. Bazi. Two-branch neural network for learning multi-label classification in uav imagery. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 2443–2446, 2019.
- [136] A. Alshehri, Y. Bazi, N. Ammour, and N. Alajlan. Multi-label classification of remote sensing imagery with deep neural networks. In *Mediterranean and Middle-East Geoscience and Remote Sensing Symposium (M2GARSS)*, pages 97–100, 2020.
- [137] G. Sumbul and B. Demir. A deep multi-attention driven approach for multi-label remote sensing image classification. *IEEE Access*, 8:95934–95946, 2020.
- [138] I. Shendryk, Y. Rist, R. Lucas, P. Thorburn, and C. Ticehurst. Deep learning-a new approach for multi-label scene classification in PlanetScope and Sentinel-2 imagery. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1116–1119, 2018.
- [139] Z. Shao, W. Zhou, X. Deng, M. Zhang, and Q. Cheng. Multilabel remote sensing image retrieval based on fully convolutional network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:318–328, 2020.
- [140] M. Topçu, A. Dede, S. Eken, and A. Sayar. Multilabel remote sensing image classification with capsule networks. In *International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, pages 1–3, 2020.
- [141] Y. Hua, L. Mou, and X. X. Zhu. Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional LSTM network for multi-label aerial image classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 149:188–199, 2019.
- [142] Y. Hua, L. Mou, and X. X. Zhu. Relation network for multilabel aerial image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 58(7):4558–4572, 2020.
- [143] N. Khan, U. Chaudhuri, B. Banerjee, and S. Chaudhuri. Graph convolutional network for multi-label VHR remote sensing scene recognition. *Neurocomputing*, 357:36–46, 2019.
- [144] J. Kang, R. Fernandez-Beltran, D. Hong, J. Chanussot, and A. Plaza. Graph relation network: Modeling relations between scenes for multilabel remote-sensing image classification and retrieval. *IEEE Transactions on Geoscience and Remote Sensing*, 59(5):4355–4369, 2020.
- [145] J. Ji, W. Jing, G. Chen, J. Lin, and H. Song. Multi-label remote sensing image classification with latent semantic dependencies. *Remote Sensing*, 12(7):1110, 2020.
- [146] Y. Diao, J. Chen, and Y. Qian. Multi-label remote sensing image classification with deformable convolutions and graph neural networks. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 521–524, 2020.
- [147] R. Huang, F. Zheng, and W. Huang. Multilabel remote sensing image annotation with multiscale attention and label correlation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:6951–6961, 2021.
- [148] J. Zhang, J. Zhang, T. Dai, and Z. He. Exploring weighted dual graph regularized non-negative matrix tri-factorization based collaborative filtering framework for multi-label annotation of remote sensing images. *Remote Sensing*, 11(8):922, 2019.
- [149] Q. Tan, Y. Liu, X. Chen, and G. Yu. Multi-label classification based on low rank representation for image annotation. *Remote Sensing*, 9(2):109, 2017.
- [150] Y. Hua, S. Lobry, L. Mou, D. Tuia, and X. X. Zhu. Learning multi-label aerial image classification under label noise: A regularization approach using word embeddings. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 525–528, 2020.

Bibliography

- [151] K. Chen, P. Jian, Z. Zhou, J. Guo, and D. Zhang. Semantic annotation of high-resolution remote sensing images via gaussian process multi-instance multilabel learning. *IEEE Geoscience and Remote Sensing Letters*, 10(6):1285–1289, 2013.
- [152] Y. Li, R. Chen, Y. Zhang, and H. Li. A CNN-GCN framework for multi-label aerial image scene classification. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1353–1356, 2020.
- [153] X. Wang, X. Xiong, and C. Ning. Multi-label remote sensing scene classification using multi-bag integration. *IEEE Access*, 7:120399–120410, 2019.
- [154] M. Salah, A. Mitiche, and I. Ayed. Multiregion image segmentation by parametric kernel graph cuts. *IEEE Transactions on Image Processing*, 20(2):545–557, 2010.
- [155] L. Ding, H. Tang, and L. Bruzzone. LANet: Local attention embedding to improve the semantic segmentation of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 59(1):426–435, 2020.
- [156] H. Li, K. Qiu, L. Chen, X. Mei, L. Hong, and C. Tao. SCAAttNet: Semantic segmentation network with spatial and channel attention mechanism for high-resolution remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, 18(5):905–909, 2020.
- [157] L. Mou and X. X. Zhu. RiFCN: Recurrent network in fully convolutional network for semantic segmentation of high resolution remote sensing images. *arXiv:1805.02091*, 2018.
- [158] L. Mou, Y. Hua, and X. X. Zhu. A relation-augmented fully convolutional network for semantic segmentation in aerial scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12416–12425, 2019.
- [159] C. Dechesne, P. Lassalle, and S. Lefèvre. Bayesian U-Net: Estimating uncertainty in semantic segmentation of earth observation images. *Remote Sensing*, 13(19):3836, 2021.
- [160] S. Liu, J. Cheng, L. Liang, H. Bai, and W. Dang. Light-weight semantic segmentation network for uav remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:8287–8296, 2021.
- [161] J. Castillo-Navarro, N. Audebert, A. Boulch, B. Le Saux, and S. Lefèvre. What data are needed for semantic segmentation in earth observation? In *Joint Urban Remote Sensing Event (JURSE)*, pages 1–4, 2019.
- [162] N. Audebert, B. Le Saux, and S. Lefèvre. Semantic segmentation of earth observation data using multimodal and multi-scale deep networks. In *Asian Conference on Computer Vision (ACCV)*, pages 180–196, 2016.
- [163] J. Castillo-Navarro, B. Le Saux, A. Boulch, N. Audebert, and S. Lefèvre. Semi-supervised semantic segmentation in earth observation: The minifrance suite, dataset analysis and multi-task network study. *Machine Learning*, pages 1–36, 2021.
- [164] N. Audebert, B. Le Saux, and S. Lefèvre. Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 140:20–32, 2018.
- [165] F. I. Diakogiannis, F. Waldner, P. Caccetta, and C. Wu. Resunet-a: A deep learning framework for semantic segmentation of remotely sensed data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 162:94–114, 2020.
- [166] D. Marcos, M. Volpi, B. Kellenberger, and D. Tuia. Land cover mapping at very high resolution with rotation equivariant cnns: Towards small yet accurate models. *ISPRS journal of photogrammetry and remote sensing*, 145:96–107, 2018.
- [167] R. Liu, L. Mi, and Z. Chen. AFNet: Adaptive fusion network for remote sensing image semantic segmentation. *IEEE Transactions on Geoscience and Remote Sensing*, 59(9), 2021.

- [168] M. Volpi and D. Tuia. Deep multi-task learning for a geographically-regularized semantic segmentation of aerial images. *ISPRS journal of photogrammetry and remote sensing*, 144:48–60, 2018.
- [169] S. Pan, Y. Tao, C. Nie, and Y. Chong. PEGNet: Progressive edge guidance network for semantic segmentation of remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, 18(4):637–641, 2020.
- [170] Y. Li, T. Shi, Y. Zhang, W. Chen, Z. Wang, and H. Li. Learning deep semantic segmentation network under multiple weakly-supervised constraints for cross-domain remote sensing image semantic segmentation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 175:20–33, 2021.
- [171] W. Li, H. Chen, and Z. Shi. Semantic segmentation of remote sensing images with self-supervised multitask representation learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:6438–6450, 2021.
- [172] G. Xia, W. Yang, Y. Delon, J. and Gousseau, H. Sun, and H. Maître. Structural high-resolution satellite image indexing. In *ISPRS Technical Commission VII Symposium*, volume 38, pages 298–303, 2010.
- [173] Q. Zou, L. Ni, T. Zhang, and Q. Wang. Deep learning based feature selection for remote sensing scene classification. *IEEE Geoscience and Remote Sensing Letters*, 12(11):2321–2325, 2015.
- [174] B. Zhao, Y. Zhong, G. Xia, and L. Zhang. Dirichlet-derived multiple topic scene classification model fusing heterogeneous features for high spatial resolution remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 54(4):2108–2123, 2016.
- [175] L. Zhao, P. Tang, and L. Huo. Feature significance-based multibag-of-visual-words model for remote sensing image scene classification. *Journal of Applied Remote Sensing*, 10(3):035004, 2016.
- [176] G. Cheng, J. Han, and X. Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017.
- [177] H. Li, X. Dou, C. Tao, Z. Wu, J. Chen, J. Peng, M. Deng, and L. Zhao. RSI-CB: A large-scale remote sensing image classification benchmark using crowdsourced data. *Sensors*, 20(6):1594, 2020.
- [178] Z. Xiao, Y. Long, D. Li, C. Wei, G. Tang, and J. Liu. High-resolution remote sensing image retrieval based on CNNs from a dimensional perspective. *Remote Sensing*, 9(7):725, 2017.
- [179] Y. Long, Y. Gong, Z. Xiao, and Q. Liu. Accurate object localization in remote sensing images based on convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 55(5):2486–2498, 2017.
- [180] P. Jin, G. Xia, F. Hu, Q. Lu, and L. Zhang. AID++: An updated version of aid on scene classification. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 4721–4724, 2018.
- [181] W. Zhou, S. Newsam, C. Li, and Z. Shao. PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval. *ISPRS Journal of Photogrammetry and Remote Sensing*, 145:197–209, 2018.
- [182] Q. Wang, S. Liu, J. Chanussot, and X. Li. Scene classification with recurrent attention of VHR remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 57(2):1155–1167, 2018.
- [183] H. Li, H. Jiang, X. Gu, J. Peng, W. Li, L. Hong, and C. Tao. CLRS: Continual learning benchmark for remote sensing image scene classification. *Sensors*, 20(4):1226, 2020.

Bibliography

- [184] X. Qi, P. Zhu, Y. Wang, L. Zhang, J. Peng, M. Wu, J. Chen, X. Zhao, N. Zang, and P. Mathiopoulos. MLRSNet: A multi-label high spatial resolution remote sensing dataset for semantic scene understanding. *ISPRS Journal of Photogrammetry and Remote Sensing*, 169:337–350, 2020.
- [185] X. X. Zhu, J. Hu, C. Qiu, Y. Shi, J. Kang, L. Mou, H. Bagheri, M. Haberle, Y. Hua, R. Huang, L. Hughes, H. Li, Y. Sun, G. Zhang, S. Han, M. Schmitt, and Y. Wang. So2Sat LCZ42: A benchmark data set for the classification of global local climate zones. *IEEE Geoscience and Remote Sensing Magazine*, 8(3):76–89, 2020.
- [186] G. Sumbul, M. Charfuelan, B. Demir, and V. Markl. BigEarthNet: A large-scale benchmark archive for remote sensing image understanding. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 5901–5904, 2019.
- [187] V. Mnih. *Machine Learning for Aerial Image Labeling*. PhD thesis, University of Toronto, 2013.
- [188] M. Volpi and V. Ferrari. Semantic segmentation of urban scenes by learning local class interactions. In *CVPRw*, 2015.
- [189] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez. Can semantic labeling methods generalize to any city? the Inria aerial image labeling benchmark. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 3226–3229, 2017.
- [190] Z. Shao, K. Yang, and W. Zhou. Performance evaluation of single-label and multi-label remote sensing image retrieval using a dense labeling dataset. *Remote Sensing*, 10(6):964, 2018.
- [191] Y. Lyu, G. Vosselman, G. Xia, A. Yilmaz, and M. Yang. UAVid: A semantic segmentation dataset for UAV imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 165:108–119, 2020.
- [192] S. Tian, A. Ma, Z. Zheng, and Y. Zhong. Hi-UCD: A large-scale dataset for urban semantic change detection in remote sensing imagery. *arXiv:2011.03247*, 2020.
- [193] A. Boguszewski, D. Batorski, N. Ziembka-Jankowska, T. Dziedzic, and A. Zambrzycka. Land-Cover.ai: Dataset for automatic mapping of buildings, woodlands, water and roads from aerial imagery. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1102–1110, 2021.
- [194] Y. Hua, L. Mou, J. Lin, K. Heidler, and X. X. Zhu. Aerial scene understanding in the wild: Multi-scene recognition via prototype-based memory networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 177:89–102, 2021.
- [195] Y. Hua, L. Mou, P. Jin, and X. X. Zhu. MultiScene: A large-scale dataset and benchmark for multi-scene recognition in single aerial images. *IEEE Transactions on Geoscience and Remote Sensing*, pages 1–13, 2021.
- [196] W. Wu, H. Qi, Z. Rong, L. Liu, and H. Su. Scribble-supervised segmentation of aerial building footprints using adversarial learning. *IEEE Access*, 6:58898–58911, 2018.
- [197] L. Maggiolo, D. Marcos, G. Moser, and D. Tuia. Improving maps from CNNs trained with sparse, scribbled ground truths using fully connected CRFs. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 2099–2102, 2018.
- [198] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Ssstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, 2012.
- [199] D. Lin, J. Dai, J. Jia, K. He, and J. Sun. ScribbleSup: Scribble-supervised convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3159–3167, 2016.

- [200] P. Krähenbühl and V. Koltun. Efficient inference in fully connected CRFs with Gaussian edge potentials. *Advances in neural information processing systems*, 24:109–117, 2011.
- [201] J. Hartigan and M. Wong. Algorithm AS 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.
- [202] A. Bearman, O. Russakovsky, V. Ferrari, and F. Li. What’s the point: Semantic segmentation with point supervision. In *European Conference on Computer Vision (ECCV)*, pages 549–565, 2016.
- [203] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2189–2202, 2012.
- [204] W. Lu, D. Gong, K. Fu, X. Sun, W. Diao, and L. Liu. Boundarymix: Generating pseudo-training images for improving segmentation with scribble annotations. *Pattern Recognition*, 117:107924, 2021.
- [205] K. Li, X. Hu, H. Jiang, Z. Shu, and M. Zhang. Attention-guided multi-scale segmentation neural network for interactive extraction of region objects from high-resolution satellite imagery. *Remote Sensing*, 12(5):789, 2020.
- [206] Y. Wei and S. Ji. Scribble-based weakly supervised deep learning for road surface extraction from remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 2021.
- [207] S. Xie and Z. Tu. Holistically-nested edge detection. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1395–1403, 2015.
- [208] V. Mnih. *Machine learning for aerial image labeling*. University of Toronto (Canada), 2013.
- [209] M. Volpi and V. Ferrari. Semantic segmentation of urban scenes by learning local class interactions. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRw)*, 2015.
- [210] K. Karalas, G. Tsagkatakis, M. Zervakis, and P. Tsakalides. Land classification using remotely sensed data: Going multilabel. *IEEE Transactions on Geoscience and Remote Sensing*, 54(6):3548–3563, 2016.
- [211] S. Koda, A. Zeggada, F. Melgani, and R. Nishii. Spatial and structured SVM for multilabel image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 56(10):5948–5960, 2018.
- [212] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. In *Advances in Neural Information Processing Systems (NIPS)*, volume 30, 2017.
- [213] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei. Relation networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3588–3597, 2018.
- [214] L. Wang, W. Li, W. Li, and L. Van Gool. Appearance-and-relation networks for video classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1430–1439, 2018.
- [215] B. Zhou, A. Andonian, and A. Torralba. Temporal relational reasoning in videos. In *European Conference on Computer Vision (ECCV)*, pages 803–818, 2018.
- [216] L. Mou, Y. Hua, and X. X. Zhu. Relation matters: Relational context-aware fully convolutional network for semantic segmentation of high resolution aerial images. *IEEE Transactions on Geoscience and Remote Sensing*, 58(11):7557–7569, 2020.
- [217] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu. Spatial transformer networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2017–2025, 2015.

Bibliography

- [218] National Research Council. *How people learn: Brain, mind, experience, and school: Expanded edition*. 2000.
- [219] E. Liu, E. Mercado III, B. Church, and I. Orduña. The easy-to-hard effect in human (*Homo sapiens*) and rat (*Rattus norvegicus*) auditory identification. *Journal of Comparative Psychology*, 122(2):132, 2008.
- [220] I. McLaren and M. Suret. Transfer along a continuum: Differentiation or association. In *Annual Conference of the Cognitive Science Society*, pages 340–345, 2000.
- [221] J. Snell, K. Swersky, and R. Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.
- [222] S. Guerriero, B. Caputo, and M. T. DeepNCM: Deep nearest class mean classifiers. In *International Conference on Learning Representations Workshop (ICLRw)*, 2018.
- [223] H. Yang, X. Zhang, F. Yin, and C. Liu. Robust classification with convolutional prototype learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3474–3482, 2018.
- [224] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. 2018.
- [225] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [226] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 4171–4186, 2019.
- [227] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. Rush. Transformers: State-of-the-art natural language processing. In *Empirical Methods in Natural Language Processing: System Demonstrations (EMNLP)*, pages 38–45, 2020.
- [228] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus. End-to-end memory networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2440–2448, 2015.
- [229] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967.
- [230] M. Sabokrou, M. Khalooei, and E. Adeli. Self-supervised representation learning via neighborhood-relational encoding. In *IEEE International Conference on Computer Vision (ICCV)*, pages 8010–8019, 2019.
- [231] Ö. Çiçek, A. Abdulkadir, S. Lienkamp, T. Brox, and O. Ronneberger. 3D U-Net: Learning dense volumetric segmentation from sparse annotation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 424–432, 2016.
- [232] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv:1301.3781*, 2013.
- [233] J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

Appendices

- A Yuansheng Hua*, Lichao Mou*, and Xiao Xiang Zhu, “Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional LSTM network for multi-label aerial image classification,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 149, pp. 188-199, 2019.
(* equal contribution)

<https://doi.org/10.1016/j.isprsjprs.2019.01.015>

Recurrently Exploring Class-wise Attention in A Hybrid Convolutional and Bidirectional LSTM Network for Multi-label Aerial Image Classification

Yuansheng Hua^{a,b,*}, Lichao Mou^{a,b,*}, Xiao Xiang Zhu^{a,b,**}

^a*Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Oberpfaffenhofen, 82234 Wessling, Germany*

^b*Signal Processing in Earth Observation (SiPEO), Technical University of Munich (TUM), Arcisstr. 21, 80333 Munich, Germany*

Abstract

This is a preprint. To read the final version please visit [e ISPRS Journal of Photogrammetry and Remote Sensing](#). Aerial image classification is of great significance in the remote sensing community, and many researches have been conducted over the past few years. Among these studies, most of them focus on categorizing an image into one semantic label, while in the real world, an aerial image is often associated with multiple labels, e.g., multiple object-level labels in our case. Besides, a comprehensive picture of present objects in a given high-resolution aerial image can provide a more in-depth understanding of the studied region. For these reasons, aerial image multi-label classification has been attracting increasing attention. However, one common limitation shared by existing methods in the community is that the co-occurrence relationship of various classes, so-called class dependency, is underexplored and leads to an inconsiderate decision. In this paper, we propose a novel end-to-end network, namely class-wise attention-based convolutional and bidirectional LSTM network (CA-Conv-BiLSTM), for this task. The proposed network consists of three indispensable components: 1) a feature extraction module, 2) a class attention learning layer, and 3) a bidirectional LSTM-based sub-network. Particularly, the feature

*The first two authors contributed equally to this work.

**Corresponding author

Email addresses: yuansheng.hua@tum.de (Yuansheng Hua), lichao.mou@dlr.de (Lichao Mou), Xiaoxiang.Zhu@dlr.de (Xiao Xiang Zhu)



Figure 1: Example high resolution aerial images with their scene labels and multiple *object* labels. Common label pairs are *highlighted*. (a) Free way: *bare soil*, ***car***, *grass*, ***pavement*** and *tree*. (b) Intersection: *building*, ***car***, *grass*, ***pavement*** and *tree*. (c) Parking lot: ***car*** and ***pavement***.

extraction module is designed for extracting fine-grained semantic feature maps, while the class attention learning layer aims at capturing discriminative class-specific features. As the most important part, the bidirectional LSTM-based sub-network models the underlying class dependency in both directions and produce structured multiple object labels. Experimental results on UCM multi-label dataset and DFC15 multi-label dataset validate the effectiveness of our model quantitatively and qualitatively.

Keywords: Multi-label Classification, High-Resolution Aerial Image, Convolutional Neural Network (CNN), Class Attention Learning, Bidirectional Long Short-Term Memory (BiLSTM), Class Dependency.

1. Introduction

With the booming of remote sensing techniques in the recent years, a huge volume of high resolution aerial imagery is now accessible and benefits a wide range of real-world applications, such as urban mapping [1, 2, 3, 4], ecological monitoring [5, 6], geomorphological analysis [7, 8, 9, 10], and traffic management [11, 12, 13]. As a fundamental bridge between aerial images and these applications, image classification, which aims at categorizing images into semantic classes, has obtained wide attention, and many researches have been conducted recently [14, 15, 16, 17, 18, 19, 20, 21, 22, 23]. However, most existing studies assume that each image belongs to only one label (e.g., scene-level labels in Fig. 1), while in reality, an image is usually associated

with multiple labels [24]. Furthermore, a comprehensive picture of objects present in an aerial image is capable of offering a holistic understanding of such image. With this intention, numerous researches, i.e., semantic segmentation [25, 26, 27] and object detection [25, 28, 29, 30], have emerged recently. Unfortunately, it is extremely labor- and time-consuming to acquire ground truths for these studies (i.e., pixel-wise segmentation masks and bounding-box-level annotations). Compared to these expensive labels, image-level labels (cf. multiple object-level labels in Fig. 1) are at a fair low cost and readily accessible. To this end, multi-label classification, aiming at assigning an image with multiple object labels, is arising in both remote sensing [31, 32, 33, 34] and computer vision communities [35, 36, 37]. In this paper, we deploy our efforts in exploring an efficient multi-label classification model.

1.1. The Challenges of Multi-label Classification

Benefited from the fast-growing remote sensing technology, large quantities of high-resolution aerial images are available and widely used in many visual tasks. Along with such huge opportunities, challenges have come up inevitably.

On one hand, it is difficult to extract high-level features from high-resolution images. Considering its complex spatial structure, conventional hand-crafted features, and mid-level semantic models [15, 38, 39, 40, 41] suffer from the poor performance of capturing holistic semantic features, which leads to an unsatisfactory classification ability.

On the other hand, underlying correlations between dependent labels are required to be unearthed for an efficient prediction of multiple object labels. E.g., the existence of ships infers to a high probable co-occurrence of the sea, while the presence of buildings is almost always accompanied by the coexistence of pavement. However, the recently proposed multi-label classification methods [31, 32, 33, 34] assumed that classes are independent and employed a set of binary classifiers [31] or a regression model [32, 33, 34] to infer the existence of each class separately.

To summarize, a well-performed multi-label classification system requires powerful capabilities of learning holistic feature representations and should be capable of harnessing the implicit class dependency.

1.2. The Motivation of Our Work

As our survey of related work shows above, recent approaches make few efforts to exploit the high-order class dependency, which constrains the performance in multi-label classification. Besides, direct utilization of CNNs pre-trained on natural image datasets [32, 33, 34] leads to a partial interpretation of aerial images due to their diverse visual patterns. Moreover, most state-of-the-art methods decompose multi-label classification into separate stages, which cuts off their inter-correlations and makes end-to-end training infeasible.

To tackle these problems, in this paper, we propose a novel end-to-end network architecture, class attention-based convolutional and bidirectional LSTM network (CA-Conv-BiLSTM), which integrates feature extraction and high-order class dependency exploitation together for multi-label classification. Contributions of our work to the literature are detailed as follows:

- We regard the multi-label classification of aerial images as a structured output problem instead of a simple regression problem. In this manner, labels are predicted in an ordered procedure, and the prediction of each label is dependent on others. As a consequence, the implicit class relevance is taken into consideration, and structured outputs are more reasonable and closer to the real-world case as compared to regression outputs.
- we propose an end-to-end trainable network architecture for multi-label classification, which consists of a feature extraction module (e.g., a modified network based on VGG-16), a class attention learning layer, and a bidirectional LSTM-based sub-network. These components are designed for extracting features from input images, learning discriminative class-specific features, and exploiting class dependencies, respectively. Besides, such a design makes it feasible to train the network in an end-to-end fashion, which enhances the compactness of our model.
- Considering that class dependencies are diverse in both directions, a bidirectional analysis is required for modeling such correlations. Therefore, we employ a bidirectional LSTM-based network, instead of a one-way recurrent neural network, to dig out class relationships.
- We build a new challenging dataset, DFC15 multi-label dataset, by reproducing from a semantic segmentation dataset, GRSS_DFC_2015

(DFC15) [42]. The proposed dataset consists of aerial images at a spatial resolution of 5 cm and can be used to evaluate the performance of networks for multi-label classification.

The following sections further introduce and discuss our network. Specifically, Section 2 provides an intuitive illustration of the class dependency and then details the structure of the proposed network in terms of its three fundamental components. Section 3 describes the setup of our experiments, and experimental results are discussed from quantitative and qualitative perspectives. Finally, the conclusion of this paper is drawn in Section 4.

2. Methodology

2.1. An Observation

Current aerial image multi-label classification methods [32, 33, 34] consider such problem as a regression issue, where models are trained to fit a binary sequence, and each digit indicates the existence of its corresponding class. Unlike one-hot vectors, a binary sequence is allowed to contain more than one 'hot' value for indicating the joint existence of multiple candidate classes in one image. Besides, several researches [31] formulate multi-label classification into several single-label classification tasks, and thus, train a set of binary classifiers for each class. Notably, one common assumption of these studies is that classes are independent of each other, and classifiers predict the existence of each category independently. However, this is violent and not accord with real life. As illustrated in Fig. 1, although images obtained in diverse scenes are assigned with multiple different labels, there are still common classes, e.g., car and pavement, coexisting in each image. This is because, in the real-life world, some classes have a strong correlation, for example, cars are often driven or parked on pavements. To further demonstrate the class dependency, we calculate conditional probabilities for each of the two categories. Let C_r denote referenced class, and C_p denote potential co-occurrence class. Conditional probability $P(C_p|C_r)$, which depicts the possibility that C_p exhibits in an image, where the existence of C_r is priorly known, can be solved with Eq. 1,

$$P(C_p|C_r) = \frac{P(C_p, C_r)}{P(C_r)}. \quad (1)$$

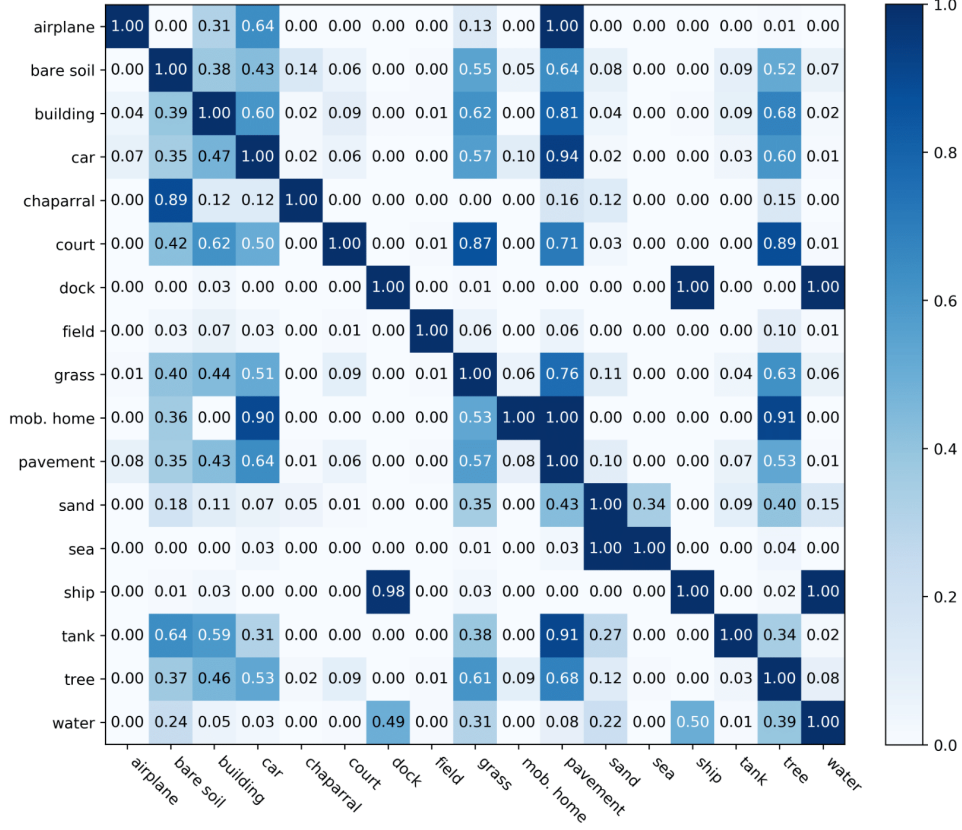


Figure 2: The co-occurrence matrix of labels in UCM multi-label dataset. Notably, all images are taken into consideration when calculating this matrix. Labels at Y-axis represent referenced classes C_r , while labels at X-axis are potential co-occurrence classes C_p . The conditional probability $P(C_p|C_r)$ of each class pair is presented in the corresponding block.

$P(C_p, C_r)$ indicates the joint occurrence probability of C_p and C_r , and $P(C_r)$ refers to the priori probability of C_r . Conditional probabilities of all class pairs in UCM multi-label datasets are listed in Fig. 2, and it is intuitive that some classes have strong dependencies. For instance, it is highly possible that there are pavements in images, which contain airplanes, buildings, cars, or tanks. Moreover, it is notable that class dependencies are not symmetric due to their particular properties. For example, $P(water|ship)$ is twice as $P(ship|water)$ due to the reason that the occurrence of ships always infer to the co-occurrence of water, while not vice versa. Therefore, to thoroughly

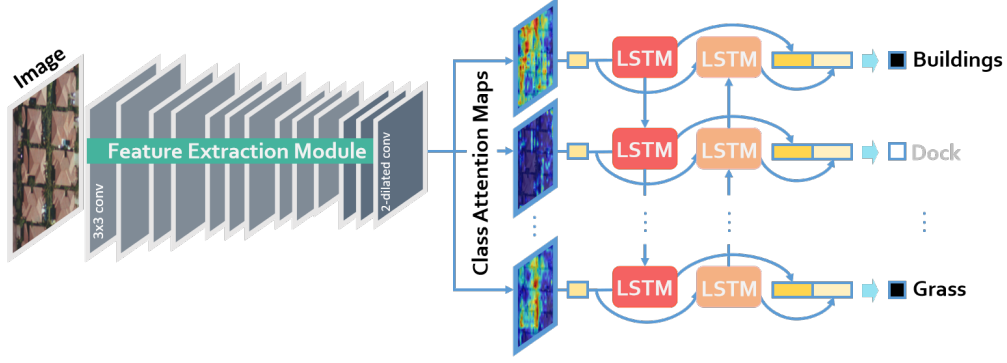


Figure 3: The architecture of the proposed CA-Conv-BiLSTM for the multi-label classification of aerial images.

dig out the correlation among various classes, it is crucial to model class probabilistic dependencies bidirectionally in a classification method.

To this end, we boil the multi-label classification down into a structured output problem, instead of a simple regression issue, and employ a unified framework of a CNN and a bidirectional RNN to 1) extract semantic features from raw images and 2) model image-label relations as well as bidirectional class dependencies, respectively.

2.2. Network Architecture

The proposed CA-Conv-BiLSTM, as illustrated in Fig. 3, is composed of three components: a feature extraction module, a class attention learning layer, and a Bidirectional LSTM-based recurrent sub-network. More specifically, the feature extraction module employs a stack of interleaved convolutional and pooling layers to extract high-level features, which are then fed into a class attention learning layer to produce discriminative class-specific features. Afterwards, a bidirectional LSTM-based recurrent sub-network is attached to model both probabilistic class dependencies and underlying relationships between image features and labels.

Section 2.2.1 details the architecture of the feature extraction module, and Section 2.2.2 describes the explicit design of the class attention learning layer. Finally, Section 2.2.3 introduces how to produce structured multi-label outputs from class-specific features via a bidirectional LSTM-based recurrent sub-network.

2.2.1. Dense High-level Feature Extraction

Learning efficient feature representations of input images is extremely crucial for the image classification task. To this end, a modern popular trend is to employ a CNN architecture to automatically extract discriminative features, and many recent studies [43, 11, 16, 44, 17, 23] have achieved great progress in a wide range of classification tasks. Inspired by this, our model adapts VGG-16 [45], one of the most welcoming CNN architectures for its effectiveness and elegance, to extract high-level features for our task.

Specifically, the feature extraction module consists of 5 convolutional blocks, and each of them contains 2 or 3 convolutional layers (as illustrated in the left of Fig. 3). Notably, the number of filters is equivalent in a common convolutional block and doubles after each pooling layer, which is utilized to reduce the spatial dimension of feature maps. The purpose of such design is to enable the feature extraction module to learn diverse features at a less computational expense. The receptive field of all convolutional filters is 3×3 , which increases nonlinearities inside the feature extraction module. Besides, the convolution stride is 1 pixel, and the spatial padding of each convolutional layer is set as 1 pixel as well. Among these convolutional blocks, max-pooling layers are interleaved for reducing the size of feature maps and meanwhile, maintaining only local representative, such as maximum in a 2×2 -pixel region. The size of pooling windows is 2×2 pixels, and the pooling stride is 2 pixels, which halves feature maps in width and length.

Features directly learned from a conventional CNN (e.g., VGG-16) are proved to be high-level and semantic, but their spatial resolution is significantly reduced, which is not favorable for generating high-dimensional class-specific features in the subsequent class attention learning layer. To address this, max-pooling layers following the last two convolutional blocks are discarded in our model, and atrous convolutional filters with dilation rate 2 are employed in the last convolutional block for preserving original receptive fields. Consequently, our feature extraction module is capable of learning high-level features with finer spatial resolution, so-called “dense”, compared to VGG-16, and it is feasible to initialize our model with pre-trained VGG-16, considering that all filters have equivalent receptive fields.

Moreover, it is noteworthy that other popular CNN architectures can be taken as prototypes of the feature extraction module, and thus, we extend researches to GoogLeNet [46] and ResNet [47] for a comprehensive evaluation of CA-Conv-BiLSTM. Regarding GoogLeNet, i.e., Inception-v3 [48],

the stride of convolutional and pooling layers after “*mixed7*” is reduced to 1 pixel, and the dilation rate of convolutional filters in “*mixed9*” is 2. For ResNet (we use ResNet-50), the convolution stride in last two residual blocks is set as 1 pixel, and the dilation rate of filters in the last residual block is 2. Besides, layers after global average pooling layers, as well as itself, are removed to ensure dense high-level feature maps.

2.2.2. Class Attention Learning Layer

Although Features extracted from pre-trained CNNs are high-level and can be directly fed into a fully connected layer for generating multi-label predictions, it is infeasible to learn high-order probabilistic dependencies by recurrently feeding it with identical features. Therefore, extracting discriminative class-wise features plays a key role in discovering class dependencies and effectively bridging CNN and RNN for multi-label classification tasks.

Here, we propose a class attention learning layer to explore features with respect to each category, and the proposed layer, illustrated in the middle of Fig. 3, consists of the following two stages: 1) generating class attention maps via a 1×1 convolutional layer with stride 1, and 2) vectorizing each class attention map to obtain class-specific features. Formally, given feature maps \mathbf{X} , extracted from the feature extraction module, with a size of $W \times W \times K$, and let \mathbf{w}_l represent the l -th convolutional filter in the class attention learning layer. The attention map \mathbf{M}_l for class l can be obtained with the following formula:

$$\mathbf{M}_l = \mathbf{X} * \mathbf{w}_l, \quad (2)$$

where l ranges from 1 to the number of classes, and $*$ represents convolution operation. Considering that the size of convolutional filters is 1×1 , a class attention map \mathbf{M}_l is intrinsically a linear combination of all channels in \mathbf{X} . With this design, the proposed class attention learning layer is capable of learning discriminative class attention maps. Some examples are shown in Fig. 4. An aerial image (cf. Fig. 4a) in UCM multi-label dataset is first fed into the feature extraction module, adapted from VGG-16, and outputs of its last convolutional block are considered as the feature maps \mathbf{X} in Eq. 2. Thus, \mathbf{X} is abundant in high-level semantic information, and the size of \mathbf{X} is $14 \times 14 \times 512$. Afterwards, a class attention learning layer, where the number of filters is equivalent to that of classes, is appended to generate class-specific feature representations with respect to all categories. With sufficient training, they are supposed to learn class-wise attention maps. It is

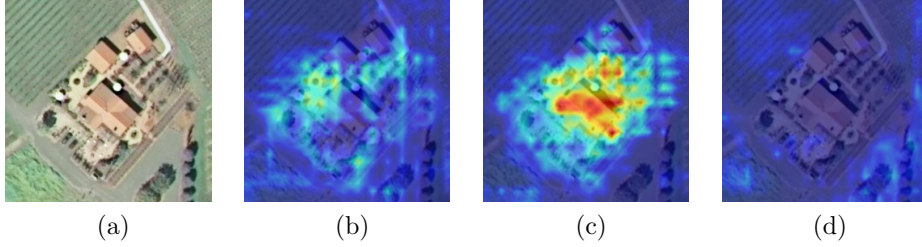


Figure 4: Example class attention maps of an a) aerial image, with respect to different classes: b) bare soil, c) building, and d) water.

observed that class attention maps highlight discriminative areas for different categories and exhibit almost no activations with respect to absent classes (as shown in Fig. 4c).

Subsequently, class attention maps \mathbf{M}_l are transformed into class-wise feature vectors \mathbf{v}_l of W^2 dimensions by vectorization. Instead of fully connecting class attention maps to each hidden unit in the following layer, we construct class-wise connections between class attention maps and their corresponding hidden units, i.e., corresponding time steps in an LSTM layer in our network. In this way, features fed into different units are retained to be class-specific discriminative and significantly contribute to the exploitation of the dynamic class dependency in the subsequent bidirectional LSTM layer.

2.2.3. Class Dependency Learning via a BiLSTM-based Sub-network

As an important branch of neural networks, RNN is widely used in dealing with sequential data, e.g., textual data and temporal series, due to its strong capabilities of exploiting implicit dependencies among inputs. Unlike CNN, RNN is characterized by its recurrent neurons, of which activations are dependent on both current inputs and previous hidden states. However, conventional RNNs suffer from the gradient vanishing problem and are found difficult to learn long-term dependencies. Therefore, in this work, we seek to model class dependencies with an LSTM-based RNN, which is first proposed in [49] and has shown great performance in processing long sequences [50, 51, 52, 53, 54].

Instead of directly summing up inputs as in a conventional recurrent layer, an LSTM layer relies on specifically designed hidden units, LSTM units, where information, such as the class dependency between category l and $l - 1$, is “memorized”, updated, and transmitted with a memory cell and

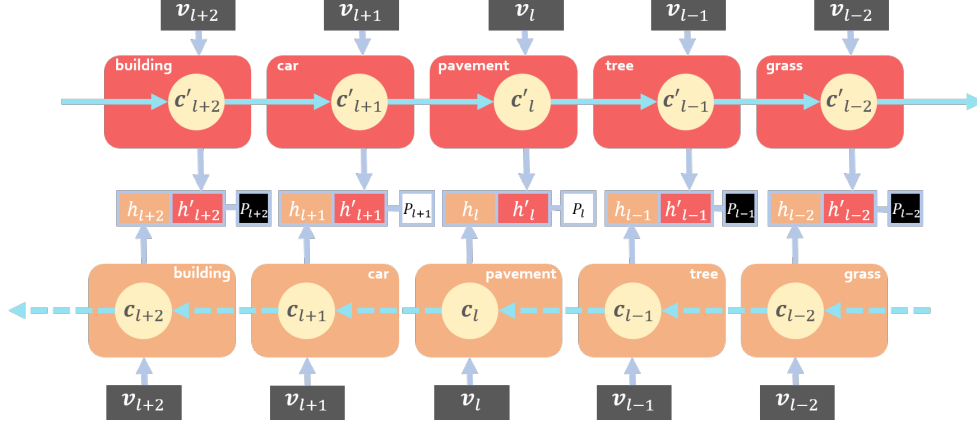


Figure 5: Illustration of the bidirectional structure. The direction of the upper stream is opposite to that of the lower stream. Notably, h'_{l-1} , c'_{l-1} denotes the activation and memory cell in the upper stream at the time step, which corresponds to class $l - 1$ for convenience (considering that the subsequent time step is usually denoted as $l + 1$).

several gates. Specifically, given a class-specific feature v_l obtained from the class attention learning layer as an input of the LSTM memory cell c_l at time step l , and let h_l represent the activation of c_l . New memory information \tilde{c}_l , learned from the previous activation h_{l-1} and the present input feature v_l , is obtained as follows:

$$\tilde{c}_l = \tanh(\mathbf{W}_{cv}v_l + \mathbf{W}_{ch}h_{l-1} + \mathbf{b}_c), \quad (3)$$

where \mathbf{W}_{cv} and \mathbf{W}_{ch} denote weight matrix from input vectors to memory cell and hidden-memory coefficient matrix, respectively, and \mathbf{b}_c is a bias term. Besides, $\tanh(\cdot)$ is the hyperbolic tangent function. In contrast to conventional recurrent units, where the \tilde{c}_l is directly used to update the current state h_l , an LSTM unit employs an input gate i_l to control the extent to which \tilde{c}_l is added, and meanwhile, partially omits uncorrelated prior information from c_{l-1} with a forget gate f_l . The two gates are performed by the following equations:

$$\begin{aligned} i_l &= \sigma(\mathbf{W}_{iv}v_l + \mathbf{W}_{ih}h_{l-1} + \mathbf{W}_{ic}c_{l-1} + \mathbf{b}_i), \\ f_l &= \sigma(\mathbf{W}_{fv}v_l + \mathbf{W}_{fh}h_{l-1} + \mathbf{W}_{fc}c_{l-1} + \mathbf{b}_f). \end{aligned} \quad (4)$$

Consequently, the memory cell c_l is updated by

$$c_l = i_l \odot \tilde{c}_l + f_l \odot c_{l-1}, \quad (5)$$

where \odot represents element-wise multiplication. Afterwards, an output gate \mathbf{o}_l , formulated by

$$\mathbf{o}_l = \sigma(\mathbf{W}_{ov}\mathbf{v}_l + \mathbf{W}_{oh}\mathbf{h}_{l-1} + \mathbf{W}_{oc}\mathbf{c}_l + \mathbf{b}_o), \quad (6)$$

is designed to determine the proportion of memory content to be exposed, and eventually, the memory cell \mathbf{c}_l at time step l is activated by

$$\mathbf{h}_l = \mathbf{o}_l \tanh(\mathbf{c}_l). \quad (7)$$

Although it is not difficult to discover that the activation of the memory cell at each time step is dependent on both input class-specific feature vectors and previous cell states. However, taking into account that the class dependency is bidirectional, as demonstrated in Section 2.1, a single-directional LSTM-based RNN is insufficient to draw a comprehensive picture of inter-class relevance. Therefore, a bidirectional LSTM-based RNN, composed of two identical recurrent streams but with reversed directions, is introduced in our model, and the hidden units are updated based on signals from not only their preceding states but also subsequent ones.

In order to practically adapt a bidirectional LSTM-based RNN to modeling the class dependency, we set the number of time steps in our bidirectional LSTM-based sub-network equivalent to that of classes under the assumption that distinct classes are predicted at respective time steps. Validated in Section 3.3 and 3.4, such design enjoys two outstanding characteristics: on one hand, the LSTM memory cell at time step l , \mathbf{c}_l , focuses on learning dependent relationship between class l and others in dual directions (cf. Fig. 5), and on the other hand, the occurrence probability of class l , P_l , can be predicted from outputs $[\mathbf{h}_l, \mathbf{h}'_l]$ with a single-unit fully connected layer:

$$P_l = \sigma(\mathbf{w}_l[\mathbf{h}_l, \mathbf{h}'_l] + \mathbf{b}_l), \quad (8)$$

where \mathbf{h}'_l denotes the activation of \mathbf{c}_l in the other direction, and σ is used as the activation function.

3. Experiments and Discussion

In this section, two high-resolution aerial datasets of different resolution used for evaluating our network are first described in Section 3.1, and then, the training strategies are introduced in Section 3.2. Afterwards, the performance of the proposed network on the two datasets is quantitatively and qualitatively evaluated in the following sections.

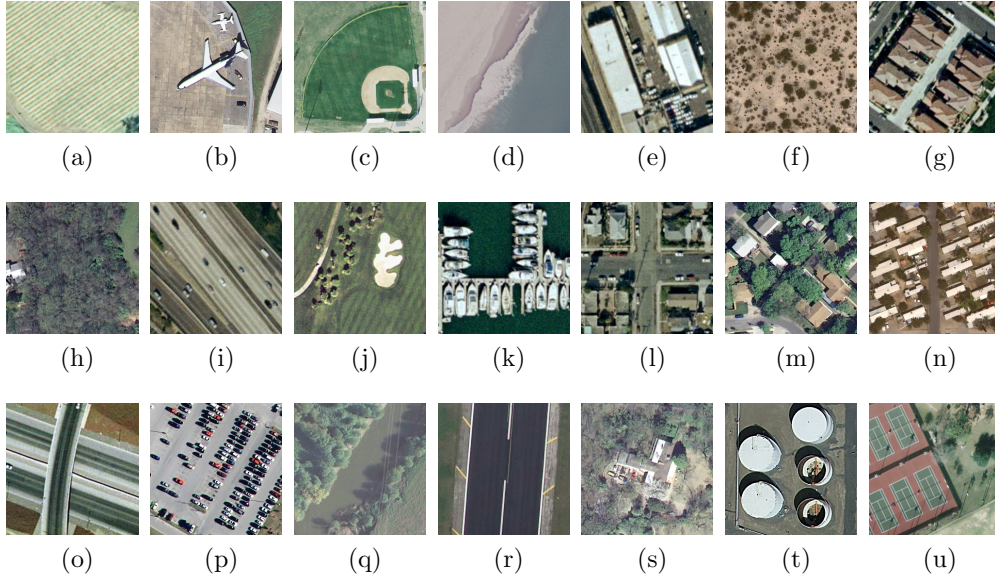


Figure 6: Example images from each scene category and their corresponding multiple *object* labels in UCM multi-label dataset. Each image is 256×256 pixels with a spatial resolution of one foot, and their scene and *object* labels are introduced: (a) Agricultural: *field* and *tree*. (b) Airplane: *airplane*, *bare soil*, *car*, *grass* and *pavement*. (c) Baseball diamond: *bare soil*, *building*, *grass*, and *pavement*. (d) Beach: *sand* and *sea*. (e) building: *building*, *car*, and *pavement*. (f) Chaparral: *bare soil* and *chaparral*. (g) Dense residential: *building*, *car*, *grass*, *pavement*, and *tree*. (h) Forest: *building*, *grass*, and *tree*. (i) Free way: *bare soil*, *car*, *grass*, *pavement*, and *tree*. (j) Golf course: *grass*, *pavement*, *sand*, and *tree*. (k) Harbor: *dock*, *ship*, and *water*. (l) Intersection: *building*, *car*, *grass*, *pavement*, and *tree*. (m) Medium residential: *building*, *car*, *grass*, *pavement*, and *tree*. (n) Mobile home park: *bare soil*, *car*, *grass*, *mobile home*, *pavement*, and *tree*. (o) Overpass: *bare soil*, *car*, and *pavement*. (p) Parking lot: *car*, *grass*, and *pavement*. (q) River: *grass*, *tree*, and *water*. (r) Runway: *grass* and *pavement*. (s) Sparse residential: *bare soil*, *building*, *car*, *grass*, *pavement*, and *tree*. (t) Storage tank: *bare soil*, *pavement*, and *tank*. (u) Tennis court: *bare soil*, *court*, *grass*, and *tree*.

3.1. Data description

3.1.1. UCM Multi-label Dataset

UCM multi-label dataset [55] is reproduced from UCM dataset [15] by reassigning them with multiple object labels. Specifically, UCM dataset consists of 2100 aerial images of 256×256 pixels, and each of them is categorized into one of 21 scene labels: airplane, beach, agricultural, baseball diamond, building, tennis courts, dense residential, forest, freeway, golf course, mobile

home park, harbor, intersection, storage tank, medium residential, overpass, sparse residential, parking lot, river, runway, and chaparral. For each of them, there are 100 images with a spatial resolution of one foot collected by cropping manually from aerial ortho imagery provided by the United States Geological Survey (USGS) National Map.

In contrast, images in UCM multi-label dataset are relabeled by assigning each image sample with one or more labels based on their primitive objects. The total number of newly defined object classes is 17: airplane, sand, pavement, building, car, chaparral, court, tree, dock, tank, water, grass, mobile home, ship, bare soil, sea, and field. It is notable that several labels, namely, airplane, building, and tank, are defined in both datasets but with variant level. In UCM dataset, they are scene-level labels, since they are predominant objects in an image and used to depict the whole image, while in UCM multi-label dataset, they are object-level labels, regarded as candidate objects in a scene. The numbers of images related to each object category are listed in Table 1, and examples from each scene category are shown in Fig. 6, as well as their corresponding object labels. To train and test our network on UCM multi-label dataset, we select 80% of sample images evenly from each scene category for training and the rest as the test set.

3.1.2. DFC15 Multi-label Dataset

Considering that images collected from the same scene may share similar patterns, alleviating task challenges, we build a new multi-label dataset, DFC15 multi-label dataset, based on a semantic segmentation dataset, DFC15 [42], which was published and first used in 2015 IEEE GRSS Data Fusion Contest. DFC15 dataset is acquired over Zeebrugge with an airborne sensor, which is 300m off the ground. In total, 7 tiles are collected in DFC dataset, and each of them is 10000×10000 pixels with a spatial resolution of 5 cm. Unlike UCM dataset, where images are assigned with image-level labels, all tiles in DFC15 dataset are labeled in pixel-level, and each pixel is categorized into 8 distinct object classes: impervious, water, clutter, vegetation, building, tree, boat, and car. Notably, vegetation refers to low vegetation, such as bushes and grasses, and has no overlap with trees. Impervious indicates impervious surfaces (e.g., roads) excluding building rooftops.

Considering our task, the following processes are conducted: First, we crop large tiles into images of 600×600 pixels with a 200-pixel-stride sliding window. Afterwards, images containing unclassified pixels are ignored, and labels of all pixels in each image are aggregated into image-level multi-

Table 1: The Number of Images in Each Object Class

Class No.	Class Name	Total	Training	Test
1	airplane	100	80	20
2	bare soil	718	577	141
3	building	691	555	136
4	car	886	722	164
5	chaparral	115	82	33
6	court	105	84	21
7	dock	100	80	20
8	field	104	79	25
9	grass	975	804	171
10	mobile home	102	82	20
11	pavement	1300	1047	253
12	sand	294	218	76
13	sea	100	80	20
14	ship	102	80	22
15	tank	100	80	20
16	tree	1009	801	208
17	water	203	161	42
-	All	2100	1680	420

labels. An important characteristic of images in DFC15 multi-label dataset is lower inter-image similarity due to that they are cropped from vast regions consecutively without specific preferences, e.g., seeking images belonging to a specific scene. Moreover, extremely high resolution makes it more challenging as compared to UCM multi-label dataset. The numbers of images containing each object label are listed in Table 2, and example images with their image-level object labels are shown in Fig. 7. To conduct the evaluation, 80% of images are randomly selected as the training set, while the others are utilized to test our network.

3.2. Training details

The proposed CA-Conv-BiLSTM is initialized with separate strategies with respect to three dominant components: 1) the feature extraction mod-

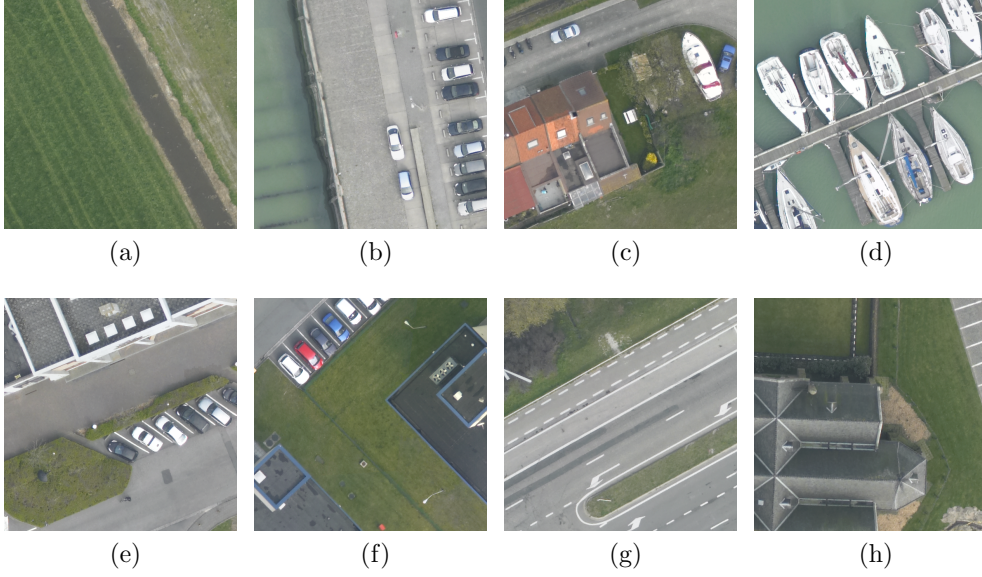


Figure 7: Example images in DFC15 multi-label dataset and their multiple *object* labels. Each image is 600×600 pixels with a spatial resolution of 5 cm. (a) *Water* and *vegetation*. (b) *Impervious*, *water*, and *car*. (c) *Impervious*, *water*, *vegetation*, *building*, and *car*. (d) *Water*, *clutter*, and *boat*. (e) *Impervious*, *vegetation*, *building*, and *car*. (f) *Impervious*, *vegetation*, *building*, and *car*. (g) *Impervious*, *vegetation*, and *tree*. (h) *Impervious*, *vegetation*, and *building*.

Table 2: The Number of Images in Each Object Class

Class No.	Class Name	Total	Training	Test
1	impervious	3133	2532	602
2	water	998	759	239
3	clutter	1891	1801	90
4	vegetation	1086	522	562
5	building	1001	672	330
6	tree	258	35	223
7	boat	270	239	31
8	car	705	478	277
-	All	3342	2674	668

ule is initialized with CNNs pre-trained on ImageNet dataset [56], 2) convolutional filters in the class attention learning layer is initialized with a Glorot uniform initializer, and 3) all weights in the bidirectional 2048-d LSTM layer are randomly initialized in the range of $[-0.1, 0.1]$ with a uniform distribution. Notably, weights in the feature extraction module are trainable and fine-tuned during the training phase of our network.

Regarding the optimizer, we chose Adam with Nesterov momentum [57], claimed to converge faster than stochastic gradient descent (SGD), and set parameters of the optimizer as recommended: $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e - 08$. The learning rate is set as $1e - 04$ and decayed by 0.1 when the validation accuracy is saturated. The loss of the network is defined as the binary cross entropy. We implement the network on TensorFlow and train it on one NVIDIA Tesla P100 16GB GPU for 100 epochs. The size of the training batch is 32 as a trade-off between GPU memory capacity and training speed. To avoid overfitting, we stop training procedure when the loss fails to decrease in five epochs. Concerning ground truths, multiple labels of an image are encoded into a multi-hot binary sequence, of which the length is equivalent to the number of all candidate labels. For each digit, 1 indicates the existence of its corresponding label, while 0 denotes the absent label.

3.3. Results on UCM Multi-label Dataset

3.3.1. Quantitative Results

To evaluate the performance of CA-Conv-BiLSTM for multi-label classification of high resolution aerial imagery, we calculate both F_1 [58] and F_2 [59] score as follows:

$$F_\beta = (1 + \beta^2) \frac{p_e r_e}{\beta^2 p_e + r_e}, \quad \beta = 1, 2, \quad (9)$$

where p_e is the example-based precision [60] of predicted multiple labels, and r_e indicates the example-based recall. They are computed by:

$$p_e = \frac{TP_e}{TP_e + FP_e}, \quad r_e = \frac{TP_e}{TP_e + FN_e}, \quad (10)$$

where TP_e , FP_e , and FN_e indicate the numbers of positive labels, which are predicted correctly (true positives) and incorrectly (false positives), and negative labels, which are incorrectly predicted (false negatives) in an example

(i.e., an image with multiple object labels in our case), respectively. Then, the average of F_2 scores of each example is formed to assess the overall accuracy of multi-label classification tasks. Besides, example-based mean precision as well as mean recall are calculated to assess the performance from the perspective of examples, while label-based mean precision and mean recall can help us understand the performance of the network from the perspective of object labels:

$$p_l = \frac{TP_l}{TP_l + FP_l}, \quad r_l = \frac{TP_l}{TP_l + FN_l}, \quad (11)$$

where TP_l , FP_l , and FN_l represent the numbers of correctly predicted positive images, incorrectly predicted positive images, and incorrectly predicted negative images with respect to each label.

For a fair validation of CA-Conv-BiLSTM, we decompose the evaluation into two components: we compare 1) CA-Conv-LSTM with standard CNNs to validate the effectiveness of employing LSTM-based recurrent sub-network, and 2) CA-Conv-BiLSTM with CA-Conv-LSTM for further assess the significance of the bidirectional structure. The detailed configurations of these competitors are listed in Table 3. For standard CNNs, we substitute last softmax layers, which are designed for single-label classification, with sigmoid layers to predict multi-hot binary sequences, where each digit indicates the probability of the presence of its corresponding category. To calculate evaluation metrics, we binarize outputs of all models with a threshold of 0.5 for producing binary sequences. Besides, our model is compared with a relevant existing method [32] for a comprehensive evaluation of its performance.

Table 4 exhibits results on UCM multi-label dataset, and it can be seen that compared to directly applying standard CNNs to multi-label classification, CA-Conv-LSTM framework performs superiorly as expected due to taking class dependencies into consideration. CA-VGG-LSTM increases the mean F_1 score by 1.03% with respect to VGGNet, while for CA-ResNet-LSTM, an increment of 1.68%, is obtained compared to ResNet. Mostly enjoying this framework, CA-GoogLeNet-LSTM achieves the best mean F_1 score of 81.78% and an increment of 1.10% in comparison with other CA-Conv-LSTM models and GoogLeNet, respectively. Moreover, CA-ResNet-LSTM shows an improvement of 3.08% of the mean F_2 score in comparison with ResNet, while CA-GoogLeNet-LSTM obtains the best F_2 score of 85.16%. To summarize, all comparisons demonstrate that instead of directly using a standard CNN as a regression task, exploiting class dependencies

Table 3: Configurations of CA-Conv-LSTM Architectures

Model	CNN model	Class Attention Map	Bi.
CA-VGG-LSTM	VGG-16	$28 \times 28 \times N$	\times
CA-VGG-BiLSTM	VGG-16	$28 \times 28 \times N$	\checkmark
CA-GoogLeNet-LSTM	Inception-v3	$17 \times 17 \times N$	\times
CA-GoogLeNet-BiLSTM	Inception-v3	$17 \times 17 \times N$	\checkmark
CA-ResNet-LSTM	ResNet-50	$28 \times 28 \times N$	\times
CA-ResNet-BiLSTM	ResNet-50	$28 \times 28 \times N$	\checkmark

N indicates the number of classes in the dataset.

Bi. indicates whether the model is bidirectional or not.

plays a key role in multi-label classification.

Concerning the signification of employing a bidirectional structure, CA-Conv-BiLSTM performs better than CA-Conv-LSTM in the mean F_1 score, and compared to Conv-RBFNN, our models achieve higher mean F_1 and F_2 scores, increased by at most 0.98% and 2.80%, respectively. Another important observation is that our proposed model is equipped with higher example-based recall but lower example-based precision, which leads to a relatively higher mean F_2 score. Notably, the F_2 score is an evaluation index used in Kaggle Amazon contest [59] to assess the performance of recognizing challenging rare objects in aerial images, and a higher score indicates a stronger capability. Table 5 exhibits several example predictions in UCM multi-label dataset. Although our model successfully predicts most multiple object labels, it is observed that the grass and tree are prone to be misclassified due to their analogous appearances. In the 4th image, the grass is a false positive when there exist trees, while in the 5th image, the tree is a false positive when the grass presents. Likewise, the bare soil in the 5th image is neglected unfortunately for its similar visual patterns with the grass.

3.3.2. Qualitative Results

In addition to validate classification capabilities of the network by computing the mean F_2 score, we further explore the effectiveness of class-specific features learned from the proposed class attention learning layer and try to “open” the black box of our network by feature visualization. Example

Table 4: Quantitative Results on UCM Multi-label Dataset (%)

Model	m. F_1	m. F_2	m. P_e	m. R_e	m. P_l	m. R_l
VGGNet [45]	78.54	80.17	79.06	82.30	86.02	80.21
VGG-RBFNN [32]	78.80	81.14	78.18	83.91	81.90	82.63
CA-VGG-LSTM	79.57	80.75	80.64	82.47	87.74	75.95
CA-VGG-BiLSTM	79.78	81.69	79.33	83.99	85.28	76.52
GoogLeNet [46]	80.68	82.32	80.51	84.27	87.51	80.85
GoogLeNet-RBFNN [32]	81.54	84.05	79.95	86.75	86.19	84.92
CA-GoogLeNet-LSTM	81.78	85.16	78.52	88.60	86.66	85.99
CA-GoogLeNet-BiLSTM	81.82	84.41	79.91	87.06	86.29	84.38
ResNet-50 [47]	79.68	80.58	80.86	81.95	88.78	78.98
ResNet-RBFNN [32]	80.58	82.47	79.92	84.59	86.21	83.72
CA-ResNet-LSTM	81.36	83.66	79.90	86.14	86.99	82.24
CA-ResNet-BiLSTM	81.47	85.27	77.94	89.02	86.12	84.26

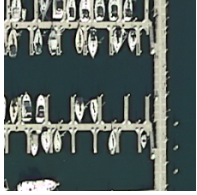



m. F_1 and m. F_2 indicate the mean F_1 and F_2 score.


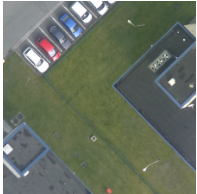
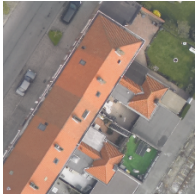
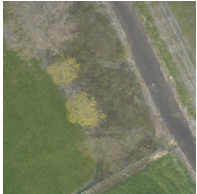

m. P_e and m. R_e indicate mean example-based precision and recall.

m. P_l and m. R_l indicate mean label-based precision and recall.

class attention maps produced by the proposed network on UCM multi-label dataset are shown in Fig. 8, where column (a) is original images, and columns (b)-(i) are class attention maps for different objects: (b) bare soil, (c) building, (d) car, (e) court, (f) grass, (g) pavement, (h) tree, and (i) water. As we can see, these maps highlight discriminative regions for positive classes, while present almost no activations when corresponding objects are absent in original images. For example, object labels of the image at the first row in Fig. 8 are building, grass, pavement, and tree, and its class attention maps for these categories are strongly activated. From images at the fourth row of Fig. 8, it can be seen that regions of the grassland, forest, and river are highlighted in their corresponding class attention maps, leading to positive predictions, while no discriminative areas are intensively activated in the other maps.

Table 5: Example Predictions on UCM and DFC15 Multi-label Dataset

Images in UCM Multi-label Dataset					
Ground Truths	dock, ship, and water	building, car, pavement, and tree	building, court, pavement, grass, and tree	car, pavement, mobile-home, and tree	bare soil, car, grass, and pavement
Predictions	dock, ship, and water	building, car, pavement, and tree	building, court, pavement, grass, and tree	car, pavement, mobile-home, tree and grass	bare soil, car, grass, tree, and pavement

Images in DFC15 Multi-label Dataset					
Ground Truths	impervious, water, and building	impervious, vegetation, car, and building	impervious, vegetation, building, clutter, and car	water, vegetation, tree	impervious, vegetation, building, car, and tree
Predictions	impervious, water, and building	impervious, vegetation, car, and building	impervious, vegetation, building, clutter, and car	impervious, water, tree, vegetation	impervious, vegetation, building, car, and tree

Red predictions indicate false positives, while blue predictions are false negatives.

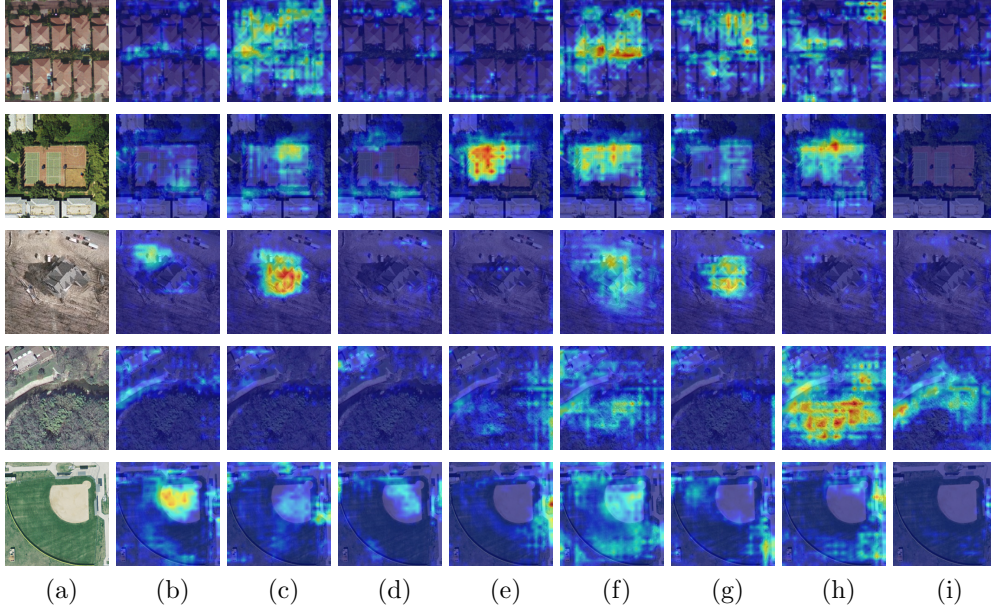


Figure 8: Example class attention maps of (a) images in UCM multi-label dataset with respect to (b) bare soil, (c) building, (d) car, (e) court, (f) grass, (g) pavement, (h) tree, and (i) water. Red indicates strong activations, while blue represents non-activations. Besides, normalization is performed based on each row for a fair comparison among class attention maps of the same images.

3.4. Results on DFC15 Multi-label Dataset

3.4.1. Quantitative Results

Following the evaluation on UCM multi-label dataset, we assess our network on DFC15 multi-label dataset by calculating the mean F_1 and F_2 score as well as mean example- and label-based precision and recall. Table 6 shows experimental results on this dataset, and the conclusion can be drawn that modeling class dependencies with a bidirectional structure contributes significantly to multi-label classification. Specifically, the mean F_1 score achieved by CA-ResNet-BiLSTM is 4.87% and 5.55% higher than CA-ResNet-LSTM and ResNet, respectively. CA-VGG-BiLSTM obtains the best mean F_1 score of 76.25% in comparison with VGGNet and CA-VGG-LSTM, and the mean F_1 score of CA-GoogLeNet-BiLSTM is 78.25%, higher than its competitors. In comparison with Conv-RBFNN, CA-Conv-BiLSTM exhibits an improvement of at most 5.29% and 4.18% in terms of the mean F_1 and F_2 score,

Table 6: Quantitative Results on DFC15 Multi-label Dataset (%)

Model	m. F_1	m. F_2	m. P_e	m. R_e	m. P_l	m. R_l
VGGNet [45]	73.86	74.09	76.16	74.95	62.57	59.95
VGGNet-RBFNN [32]	72.21	73.02	74.08	74.42	60.82	66.58
CA-VGG-LSTM	75.46	75.85	77.95	76.95	73.56	59.19
CA-VGG-BiLSTM	76.25	76.93	78.27	78.30	74.99	64.31
GoogLeNet [46]	74.99	73.41	81.01	73.01	71.80	53.95
GoogLeNet-RBFNN [32]	73.38	72.62	78.46	72.94	64.62	63.22
CA-GoogLeNet-LSTM	75.67	75.46	79.08	76.12	70.22	60.65
CA-GoogLeNet-BiLSTM	78.25	76.80	83.97	76.52	82.98	61.04
ResNet-50 [47]	78.10	76.21	84.89	75.64	81.50	59.99
ResNet-RBFNN [32]	78.36	78.08	82.64	78.76	72.01	69.85
CA-ResNet-LSTM	78.78	76.65	85.66	75.84	83.83	60.05
CA-ResNet-BiLSTM	83.65	80.61	91.93	79.12	94.35	62.35

respectively. To conclude, all these increments demonstrate the effectiveness and robustness of our bidirectional structure for high-resolution aerial image multi-label classification. Several example predictions in DFC15 multi-label dataset are shown in Table 5. The last two examples of DFC15 multi-label dataset show that trees are false negatives with the occurrence of vegetations due to their similar appearances. Moreover, we note the best result [61] in 2015 IEEE GRSS Data Fusion Contest achieves 71.18% in the mean F1 score, which is reduced by 12.47% with respect to our best result. This is because predicting dense pixel-level labels is challenging in comparison with classifying multiple image-level labels.

3.4.2. Qualitative Results

To study the effectiveness of class-specific features, we visualize class attention maps learned from the proposed class attention learning layer, as shown in Fig. 9. Columns (b)-(i) are example class attention maps with respect to (b) impervious, (c) water, (d) clutter, (e) vegetation, (f) building, (g) tree, (h) boat, and (i) car. As we can see, figures at column (b) of Fig.

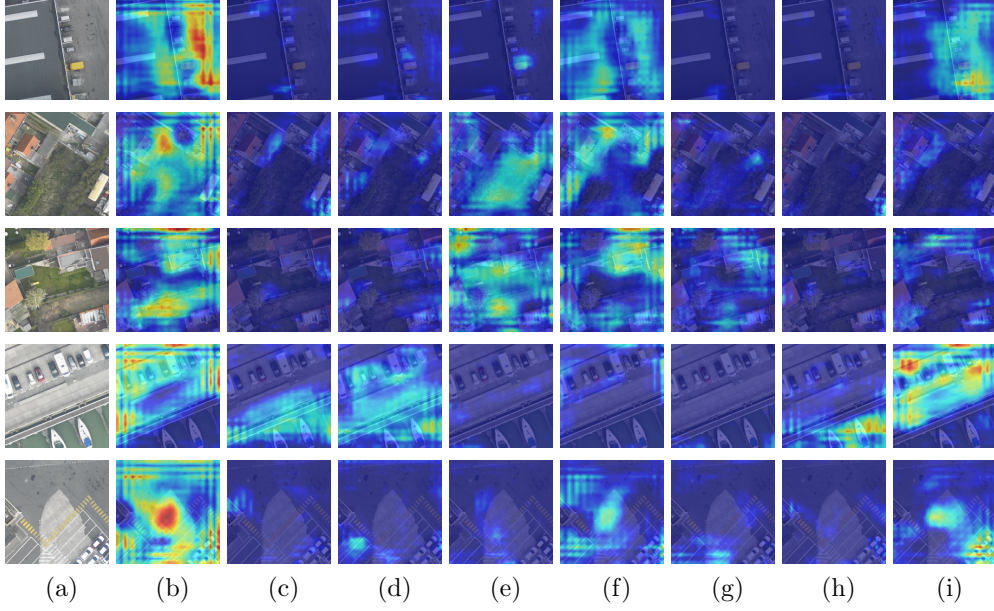


Figure 9: Example class attention maps of (a) images in DFC15 dataset with respect to (b) impervious, (c) water, (d) clutter, (e) vegetation, (f) building, (g) tree, (h) boat, and (i) car. Red indicates strong activations, while blue represents non-activations. Besides, normalization is performed based on each row for a fair comparison among class attention maps of the same images.

Fig. 9 show that the network pays high attention to impervious regions, such as parking lots, while figures at column (i) highlight regions of cars. However, some of class attention maps for negative object labels exhibit unexpected strong activations. For instance, the class attention map for the car at the third row of Fig. 9 is not supposed to highlight any region due to its absence of cars. This can be explained as the highlighted regions share similar patterns as cars, which also illustrates why the network made wrong predictions (cf. wrongly predicted car label in Fig. 9). Overall, the visualization of class attention maps demonstrates that the features captured from the proposed class attention learning layer are discriminative and class-specific. Besides, we note that there exist strong border artifacts in figures, especially those at column (b) of Fig. 9, which questions whether improving the quality of class attention maps benefits the effectiveness of the BiLSTM-based sub-network. Then we experimented with using the skip connection scheme in order to

refine class attention maps. Experimental results demonstrated that this provides negligible improvements.

4. Conclusion

In this paper, we propose a novel network, CA-Conv-BiLSTM, for the multi-label classification of high-resolution aerial imagery. The proposed network is composed of three indispensable elements: 1) a feature extraction module, 2) a class attention learning layer, and 3) a bidirectional LSTM-based sub-network. Specifically, the feature extraction module is responsible for capturing fine-grained high-level feature maps from raw images, while the class attention learning layer is designed for extracting discriminative class-specific features. Afterwards, the bidirectional LSTM-based sub-network is used to model the underlying class dependency in both directions and predict multiple object labels in a structured manner. With such design, the prediction of multiple object-level labels is performed in an ordered procedure, and outputs are structured sequences instead of discrete values. We evaluate our network on two datasets, UCM multi-label dataset and DFC15 multi-label dataset, and experimental results validate the effectiveness of our model from both quantitative and qualitative respects. On one hand, the mean F_2 score is increased by at most 0.0446 compared to other competitors. On the other hand, visualized class attention maps, where discriminative regions for existing objects are strongly activated, demonstrate that features learned from this layer are class-specific and discriminative. Looking into the future, the application of our network can be extended to fields, such as weakly supervised semantic segmentation and object localization.

Acknowledgment

This work is jointly supported by the China Scholarship Council, the Helmholtz Association under the framework of the Young Investigators Group SiPEO (VH-NG-1018, www.sipeo.bgu.tum.de), and the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No. ERC-2016-StG-714087, Acronym: *So2Sat*). In addition, the authors would like to thank the National Center for Airborne Laser Mapping and the Hyperspectral Image Analysis Laboratory at the University of Houston for acquiring and providing the data used in this study, and the IEEE GRSS Image Analysis and Data Fusion Technical Committee.

References

- [1] D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Datcu, U. Stilla, Classification with an edge: Improving semantic image segmentation with boundary detection, *ISPRS Journal of Photogrammetry and Remote Sensing* 135 (January) (2018) 158–172.
- [2] N. Audebert, B. L. Saux, S. Lefèvre, Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks, *ISPRS Journal of Photogrammetry and Remote Sensing* 140 (June) (2018) 20–32.
- [3] D. Marcos, M. Volpi, B. Kellenberger, D. Tuia, Land cover mapping at very high resolution with rotation equivariant CNNs: Towards small yet accurate models, *ISPRS Journal of Photogrammetry and Remote Sensing*.
- [4] L. Mou, X. X. Zhu, RiFCN: Recurrent network in fully convolutional network for semantic segmentation of high resolution remote sensing images, *arXiv:1805.02091*.
- [5] P. Zarco-Tejada, R. Diaz-Varela, V. Angileri, P. Loudjani, Tree height quantification using very high resolution imagery acquired from an unmanned aerial vehicle (UAV) and automatic 3D photo-reconstruction methods, *European Journal of Agronomy* 55 (2014) 89–99.
- [6] D. Wen, X. Huang, H. Liu, W. Liao, L. Zhang, Semantic classification of urban trees using very high resolution satellite imagery, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 10 (4) (2017) 1413–1424.
- [7] L. Mou, X. X. Zhu, IM2HEIGHT: Height estimation from single monocular imagery via fully residual convolutional-deconvolutional network, *arXiv:1802.10249*.
- [8] S. Lucchesi, M. Giardino, L. Perotti, Applications of high-resolution images and DTMs for detailed geomorphological analysis of mountain and plain areas of NW Italy, *European Journal of Remote Sensing* 46 (1) (2013) 216–233.
- [9] Q. Weng, Z. Mao, J. Lin, X. Liao, Land-use scene classification based on a CNN using a constrained extreme learning machine, *International Journal of Remote Sensing* 0 (0) (2018) 1–19.

- [10] G. Cheng, J. Han, X. Lu, Remote sensing image scene classification: Benchmark and state of the art, *Proceedings of the IEEE* 105 (10) (2017) 1865–1883.
- [11] L. Mou, X. X. Zhu, Vehicle instance segmentation from aerial image and video using a multi-task learning residual fully convolutional network, *IEEE Transactions on Geoscience and Remote Sensing*.
- [12] L. Mou, X. X. Zhu, Spatiotemporal scene interpretation of space videos via deep neural network and tracklet analysis, in: *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2016.
- [13] Q. Li, L. Mou, Q. Liu, Y. Wang, X. X. Zhu, Hsf-net: Multi-scale deep feature embedding for ship detection in optical remote sensing imagery, *IEEE Transactions on Geoscience and Remote Sensing*.
- [14] K. Nogueira, O. Penatti, J. dos Santos, Towards better exploiting convolutional neural networks for remote sensing scene classification, *Pattern Recognition* 61 (2017) 539–556.
- [15] Y. Yang, S. Newsam, Bag-of-visual-words and spatial extensions for land-use classification, in: *International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*, 2010.
- [16] G. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, X. Lu, AID: A benchmark data set for performance evaluation of aerial scene classification, *IEEE Transactions on Geoscience and Remote Sensing* 55 (7) (2017) 3965–3981.
- [17] X. X. Zhu, D. Tuia, L. Mou, S. Xia, L. Zhang, F. Xu, F. Fraundorfer, Deep learning in remote sensing: A comprehensive review and list of resources, *IEEE Geoscience and Remote Sensing Magazine* 5 (4) (2017) 8–36.
- [18] B. Demir, L. Bruzzone, Histogram-based attribute profiles for classification of very high resolution remote sensing images, *IEEE Transactions on Geoscience and Remote Sensing* 54 (4) (2016) 2096–2107.
- [19] F. Hu, G. Xia, J. Hu, L. Zhang, Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery, *Remote Sensing* 7 (11) (2015) 14680–14707.

- [20] F. Hu, G. Xia, Y. W., Z. L., Recent advances and opportunities in scene classification of aerial images with deep models, in: IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2018.
- [21] F. Zhang, B. Du, L. Zhang, Saliency-guided unsupervised feature learning for scene classification, IEEE Transactions on Geoscience and Remote Sensing 53 (4) (2015) 2175–2184.
- [22] X. Huang, H. Chen, J. Gong, Angular difference feature extraction for urban scene classification using ZY-3 multi-angle high-resolution satellite imagery, ISPRS Journal of Photogrammetry and Remote Sensing 135 (2018) 127 – 141.
- [23] L. Mou, X. X. Zhu, M. Vakalopoulou, K. Karantzas, N. Paragios, B. Le Saux, G. Moser, D. Tuia, Multitemporal very high resolution from space: Outcome of the 2016 IEEE GRSS data fusion contest, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 10 (8) (2017) 3435–3447.
- [24] Q. Tan, Y. Liu, X. Chen, G. Yu, Multi-label classification based on low rank representation for image annotation, Remote Sensing 9 (2) (2017) 109.
- [25] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, in: Advances in Neural Information Processing Systems, 2015.
- [26] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
- [27] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: A deep convolutional encoder-decoder architecture for image segmentation, arXiv:1511.00561.
- [28] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2001.

- [29] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: IEEE conference on computer vision and pattern recognition (CVPR), 2017.
- [30] S. Ren, K. He, R. Girshick, X. Zhang, J. Sun, Object detection networks on convolutional feature maps, IEEE transactions on Pattern Analysis and Machine Intelligence 39 (7) (2017) 1476–1481.
- [31] K. Karalas, G. Tsagkatakis, M. Zervakis, P. Tsakalides, Land classification using remotely sensed data: Going multilabel, IEEE Transactions on Geoscience and Remote Sensing 54 (6) (2016) 3548–3563.
- [32] A. Zeggada, F. Melgani, Y. Bazi, A deep learning approach to UAV image multilabeling, IEEE Geoscience and Remote Sensing Letters 14 (5) (2017) 694–698.
- [33] S. Koda, A. Zeggada, F. Melgani, R. Nishii, Spatial and structured SVM for multilabel image classification, IEEE Transactions on Geoscience and Remote Sensing (2018) 1–13.
- [34] A. Zeggada, S. Benbraika, F. Melgani, Z. Mokhtari, Multilabel conditional random field classification for UAV images, IEEE Geoscience and Remote Sensing Letters 15 (3) (2018) 399–403.
- [35] G. Patterson, J. Hays, Sun attribute database: Discovering, annotating, and recognizing scene attributes, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012.
- [36] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, Y.-T. Zheng, NUS-WIDE: A real-world web image database from National University of Singapore, in: ACM Conference on Image and Video Retrieval (CIVR), 2009.
- [37] M. Everingham, L. Van Gool, C. Williams, J. Winn, A. Zisserman, The Pascal visual object classes (VOC) challenge, International Journal of Computer Vision 88 (2) (2010) 303–338.
- [38] W. Shao, W. Yang, G. Xia, G. Liu, A hierarchical scheme of multiple feature fusion for high-resolution satellite scene categorization, in: International Conference on Computer Vision Systems, 2013.

- [39] V. Risojevic, Z. Babic, Fusion of global and local descriptors for remote sensing image classification, *IEEE Geoscience and Remote Sensing Letters* 10 (4) (2013) 836–840.
- [40] D. Lowe, Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision* 60 (2) (2004) 91–110.
- [41] Q. Zhu, Y. Zhong, B. Zhao, G. S. Xia, L. Zhang, Bag-of-visual-words scene classifier with local and global features for high spatial resolution remote sensing imagery, *IEEE Geoscience and Remote Sensing Letters* 13 (6) (2016) 747–751.
- [42] 2015 IEEE GRSS data fusion contest, <http://www.grss-ieee.org/community/technical-committees/data-fusion>, online.
- [43] Y. Hua, L. Mou, X. X. Zhu, LAHNet: A convolutional neural network fusing low- and high-level features for aerial scene classification, in: *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2018.
- [44] J. Kang, Y. Wang, M. Krner, H. Taubenbck, X. X. Zhu, Building instance classification using street view images.
- [45] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, *arXiv:1409.1556*.
- [46] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [47] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [48] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [49] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Computation* 9 (8) (1997) 1735–1780.

- [50] A. Graves, Generating sequences with recurrent neural networks, arXiv:1308.0850.
- [51] F. Gers, J. Schmidhuber, F. Cummins, Learning to forget: Continual prediction with LSTM, in: International Conference on Artificial Neural Networks, 1999.
- [52] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, Y. Bengio, Show, attend and tell: Neural image caption generation with visual attention, in: International Conference on Machine Learning, 2015.
- [53] L. Mou, P. Ghamisi, X. Zhu, Deep recurrent neural networks for hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 55 (7) (2017) 3639–3655.
- [54] L. Mou, L. Bruzzone, X. X. Zhu, Learning spectral-spatial-temporal features via a recurrent convolutional neural network for change detection in multispectral imagery, arXiv:1803.02642.
- [55] B. Chaudhuri, B. Demir, S. Chaudhuri, L. Bruzzone, Multilabel remote sensing image retrieval using a semisupervised graph-theoretic method, IEEE Transactions on Geoscience and Remote Sensing 56 (2) (2018) 1144–1158.
- [56] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.
- [57] T. Dozat, Incorporating Nesterov momentum into Adam, http://cs229.stanford.edu/proj2015/054_report.pdf, online.
- [58] X. Wu, Z. Zhou, A unified view of multi-label performance measures, arXiv:1609.00288.
- [59] Planet: Understanding the Amazon from space, <https://www.kaggle.com/c/planet-understanding-the-amazon-from-space#evaluation>, online.
- [60] G. Tsoumakas, I. Vlahavas, Random k-labelsets: An ensemble method for multilabel classification, in: European Conference on Machine Learning, 2007.

- [61] M. Campos-Taberner, A. Romero-Soriano, C. Gatta, G. Camps-Valls, A. Lagrange, B. L. Saux, A. Beaupre, A. Boulch, A. Chan-Hon-Tong, S. Herbin, H. Randrianarivo, M. Ferecatu, M. Shimoni, G. Moser, D. Tuia, Processing of extremely high-resolution LiDAR and RGB data: Outcome of the 2015 IEEE GRSS data fusion contestpart A: 2-D contest, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 9 (12) (2016) 5547–5559.

B Yuansheng Hua, Lichao Mou, and
Xiao Xiang Zhu, “Relation network for
multilabel aerial image classification,”
*IEEE Transactions on Geoscience
and Remote Sensing*, vol. 58, no. 7,
pp. 4558-4572, 2020.

<https://doi.org/10.1109/TGRS.2019.2963364>

Relation Network for Multi-label Aerial Image Classification

Yuansheng Hua, Lichao Mou, and Xiao Xiang Zhu, *Senior Member, IEEE*

Abstract—This is a preprint. To read the final version please visit *IEEE Transactions on Geoscience and Remote Sensing*. Multi-label classification plays a momentous role in perceiving intricate contents of an aerial image and triggers several related studies over the last years. However, most of them deploy few efforts in exploiting label relations, while such dependencies are crucial for making accurate predictions. Although an LSTM layer can be introduced to modeling such label dependencies in a chain propagation manner, the efficiency might be questioned when certain labels are improperly inferred. To address this, we propose a novel aerial image multi-label classification network, attention-aware label relational reasoning network. Particularly, our network consists of three elemental modules: 1) a label-wise feature parcel learning module, 2) an attentional region extraction module, and 3) a label relational inference module. To be more specific, the label-wise feature parcel learning module is designed for extracting high-level label-specific features. The attentional region extraction module aims at localizing discriminative regions in these features without region proposal generation, and yielding attentional label-specific features. The label relational inference module finally predicts label existences using label relations reasoned from outputs of the previous module. The proposed network is characterized by its capacities of extracting discriminative label-wise features and reasoning about label relations naturally and interpretably. In our experiments, we evaluate the proposed model on two multi-label aerial image datasets, of which one is newly produced. Quantitative and qualitative results on these two datasets demonstrate the effectiveness of our model. To facilitate progress in the multi-label aerial image classification, our produced dataset will be made publicly available.

Index Terms—Convolutional neural network (CNN), Label relational reasoning, Attentional region extraction, Multi-label classification, High-resolution aerial image.

I. INTRODUCTION

Recent advancements of remote sensing techniques have boosted the volume of attainable high-resolution aerial images, and massive amounts of applications, such as urban cartography [1], [2], [3], [4], traffic monitoring [5], [6], [7], terrain

This work is jointly supported by the China Scholarship Council, the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (grant agreement No [ERC-2016-StG-714087], Acronym: *So2Sat*), and Helmholtz Association under the framework of the Young Investigators Group “SiPEO” (VH-NG-1018, www.sipeco.bgu.tum.de), Helmholtz Artificial Intelligence Cooperation Unit (HAICU) - Local Unit “Munich Unit @Aeronautics, Space and Transport (MASTr)” and Helmholtz Excellent Professorship “Data Science in Earth Observation - Big Data Fusion for Urban Research”. Besides, the authors would like to thank Xinyi Liu for supporting this work with data annotation. (*Corresponding author: Xiao Xiang Zhu.*)

Y. Hua, L. Mou, and X. X. Zhu are with the Remote Sensing Technology Institute, German Aerospace Center, 82234 Weßling, Germany, and also with Signal Processing in Earth Observation, Technical University of Munich, 80333 Munich, Germany (e-mail: yuansheng.hua@dlr.de; lichao.mou@dlr.de; xiaoxiang.zhu@dlr.de).

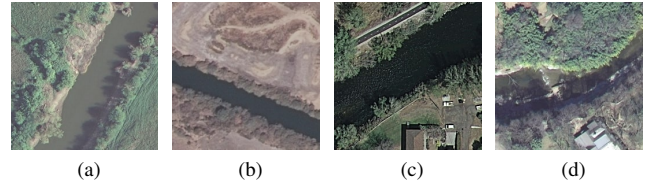


Fig. 1: Example aerial images of scene river and objects present in them. (a) bare soil, grass, tree, and water. (b) water, bare soil, and tree. (c) water, building, grass, car, tree, pavement, and bare soil. (d) water, building, grass, bare soil, tree, and sand.

surface analysis [8], [9], [10], [11], and ecological scrutiny [12], [13], have benefited from these developments. For this reason, the aerial image classification has become one of the fundamental visual tasks in the remote sensing community and drawn a plethora of research interests [14], [15], [16], [17], [18], [19], [20], [21]. The classification of aerial images refers to assigning these images with specific labels according to their semantic contents, and a common hypothesis shared by many relevant studies is that an image should be labeled with only one semantic category, such as scene categories (see Fig. 1). Although such image-level labels [22], [23] are capable of delineating images from a macroscopic perspective, it is infeasible for them to provide a comprehensive view of objects in aerial images. To tackle this, huge quantities of algorithms have been proposed to identify each pixel in an image [24], [25], [26] or localize objects with bounding boxes [27], [28], [29]. However, the acquisition of requisite ground truths (i.e., pixel-wise annotations and bounding boxes) demands enormous expertise and human labors, which makes relevant datasets expensive and difficult to access. With this intention, multi-label image classification now attracts increasing attention in the remote sensing community [30], [31], [32], [33], [34] owing to that 1) a comprehensive picture of aerial image contents can be drawn, and 2) datasets required in this task are not expensive (only image-level labels are needed).

Fig. 1 illustrates the difference between image-level scene labels and object labels. As shown in this figure, although these four images are assigned with the same scene label, their multiple object labels vary a lot. It is worth noting that the identification of some objects can actually offer important cues to understand a scene more deeply. For example, the existence of *building* and *pavement* indicates a high probability that rivers in Fig. 1c and 1d are very close to areas with frequent human activities, while rivers in Fig. 1a and 1b are more

likely in the wild due to the absence of human activity cues. In contrast, simply recognizing scene labels can hardly provide such information. Therefore, in this paper, we dedicate our efforts to explore an effective model for the multi-label classification of aerial images.

A. Challenges of Identifying Multiple labels

In identifying multiple labels of an aerial image, two main challenges need to be faced with. One is how to extract semantic feature representations from raw images. This is crucial but difficult especially for high-resolution aerial images, as they always contain complicated spatial contextual information. Conventional approaches mainly resort to manually crafted features and semantic models [22], [35], [36], [37], [38], while these methods cannot effectively extract high-level semantics and lead to a limited performance in classification [23]. Hence an efficient high-level feature extractor is desirable.

The other challenge is how to take full advantage of label correlations to infer multiple object labels of an aerial image. In contrast to single-label classification, which mainly focuses on modeling image-label relevance, exploring and modeling label-label correlations plays a supplementary yet essential role in identifying multiple objects in aerial images. For instance, the presence of ships confidently infers the co-occurrence of water or sea, while the existence of a car suggests a high probability of the appearance of pavements. Unfortunately, such label correlations are scarcely addressed in the literature. One solution is to use a recurrent neural network (RNN) to learn label dependencies. However, this is done with a chain propagation fashion, and its performance heavily depends on the learning effectiveness of its long-term memorization. Moreover, in this way, label relations are modeled implicitly, which leads to a lack of interpretability.

Overall, an efficient multi-label classification model is supposed to be capable of not only learning high-level feature representations but also modeling label correlations effectively.

B. Related Work

Zegeye and Demir [39] propose a multi-label active learning framework using a multi-label support vector machine (SVM), relying on both the multi-label uncertainty and diversity. Koda et al. [32] introduce a spatial and structure SVM for multi-label classification by considering spatial relations between a given patch and its neighbors. Similarly, Zeggada et al. [33] employ a conditional random field (CRF) framework to model spatial contextual information among adjacent patches for improving the performance of classifying multiple object labels.

With the development of computational resources and deep learning, very recent approaches mainly resort to deep networks for multi-label classification. In [31], the authors make use of a standard CNN architecture to extract feature representations and then feed them into a multi-label classification layer, which is composed of customized thresholding operations, for predicting multiple labels. In [40], the authors demonstrate that training a CNN for multi-label classification with a limited amount of labeled data usually leads to an underwhelming-performance model and propose a dynamic

data augmentation method for enlarging training sets. More recently, Sumbul and Demir [41] propose a CNN-RNN method for identifying labels in multi-spectral images, where a bidirectional LSTM is employed to model spatial relationships among image patches. In order to explore inherent correlations among object labels, [34] proposes a CNN-LSTM hybrid network architecture to learn label dependencies for classifying object labels of aerial images. Besides, we also notice that several zero short learning researches focus on employing prior knowledge to model label relations. For instance, Sumbul et al. [42] apply an unsupervised word embedding model to encoding labels into word vectors, which are supposed to contain label semantics, and then model label relationships with these vectors. Lee et al. [43] propose to learn label relations from structured knowledge graphs observed from the real world.

C. The Motivation of Our Work

In order to explicitly model label relations, we propose a label relational inference network for multi-label aerial image classification. This work is inspired by recent successes of relation networks in visual question answering [44], object detection [45], video classification [46], activity recognition in videos [47], and semantic segmentation [48]. A relation network is characterized by its inherent capability of inferring relations between an individual entity (e.g., a region in an image or a frame in a video) and all other entities (e.g., all regions in the image or all frames in the video). Besides, to increase the effectiveness of relational reasoning, we make use of a spatial transformer, which is often used to enhance the transformation invariance of deep neural networks [49], to reduce the impact of irrelevant semantic features.

More specifically, in this work, an innovative end-to-end multi-label aerial image classification network, termed as attention-aware label relational reasoning network, is proposed and characterized by its capabilities of localizing label-specific discriminative regions and explicitly modeling semantic label dependencies for the task. This paper's contributions are threefold.

- We propose a novel multi-label aerial image classification network, attention-aware label relational reasoning network, which consists of three imperative components: a label-wise feature parcel learning module, an attentional region extraction module, and a label relational inference module. To our best knowledge, it is the first time that the idea of relation networks is employed to predict multiple object labels of aerial images, and experimental results demonstrate its effectiveness.
- We extract attentional regions from the label-wise feature parcels in a proposal-free fashion. Particularly, a learnable spatial transformer is employed to localize attentional regions, which are assumed to contain discriminative information, and then re-coordinate them into a given size. By doing so, attentional feature parcels can be yielded.
- To facilitate progress in the multi-label aerial image classification, we produce a new dataset, AID multi-label

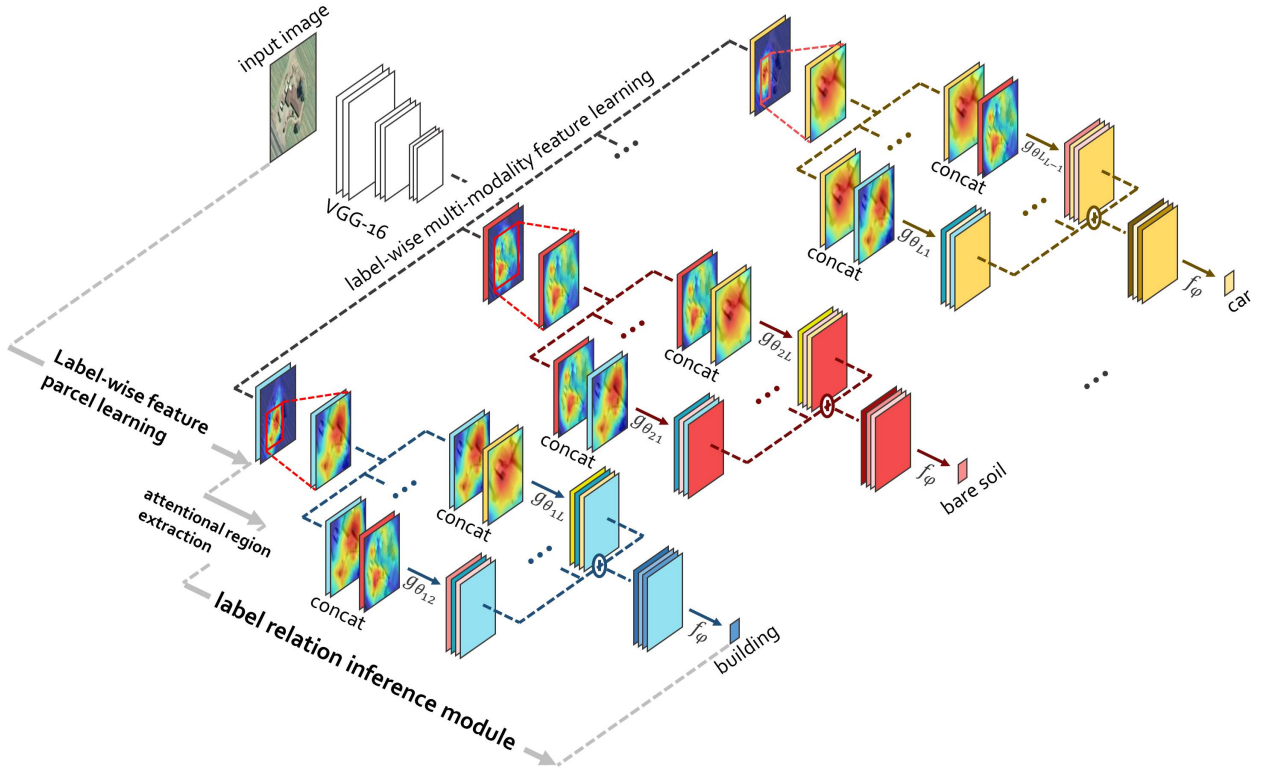


Fig. 2: The architecture of the proposed attention-aware label relational reasoning network.

dataset, by relabeling images in the AID dataset [23]. In comparison with the UCM multi-label dataset [50], the proposed dataset is more challenging due to diverse spatial resolutions of images, more scenes, and more samples.

The remaining sections of this paper are organized as follows. Section II delineates three elemental modules of our proposed network, and Section III introduces experiments, where experimental setups are given and results are analyzed and discussed. Eventually, Section IV draws a conclusion of this paper.

II. METHODOLOGY

A. Network Architecture

As illustrated in Fig. 2, the proposed network comprises three components: a label-wise feature parcel learning module, an attentional region extraction module, and a label relational inference module. Let L be the number of object labels and l be the l -th label. The label-wise feature parcel learning module is designed to extract high-level feature maps \mathbf{X}_l with K channels, termed as *feature parcel* (for more details refer to Section II-B), for each label l . The attentional region extraction module is used to localize discriminative regions in each \mathbf{X}_l and generate an attentional feature parcel \mathbf{A}_l , which is supposed to contain the most relevant semantics with respect to the label l . Finally, relations among \mathbf{A}_l and all other label-wise attentional feature parcels are reasoned about by the label

relational inference module for predicting the presence of the object l .

Details of the proposed network are introduced in the remaining sections.

B. Label-wise Feature Parcel Learning

The extraction of high-level features is crucial for visual recognition tasks, and many recent studies adopt CNNs owing to their remarkable performance in learning such features [15], [51], [52], [53], [54], [55], [56]. Hence, we take a standard CNN as the backbone of the label-wise feature parcel learning module in our model. As shown in Fig. 2, an aerial image is first fed into a CNN (e.g., VGG-16), which consists of only convolutional and max-pooling layers, for generating high-level feature maps. Subsequently, these features are encoded into L feature parcels for each label l via a label-wise multi-modality feature learning layer. To implement this layer, we first employ a convolutional layer with KL filters, whose size is 1×1 , to extract KL feature maps. Afterwards, we divide these features into L feature parcels, and each includes K feature maps. That is to say, for each label, K specific feature maps are learned, so-called *feature parcel*, to extract discriminative semantics after the end-to-end training of the whole network. We denote the feature parcel for label l as \mathbf{X}_l in the following statements.

In our experiment, we notice that \mathbf{X}_l with a higher resolution is beneficial for the subsequent module to localize discriminative regions, as more spatial contextual cues are

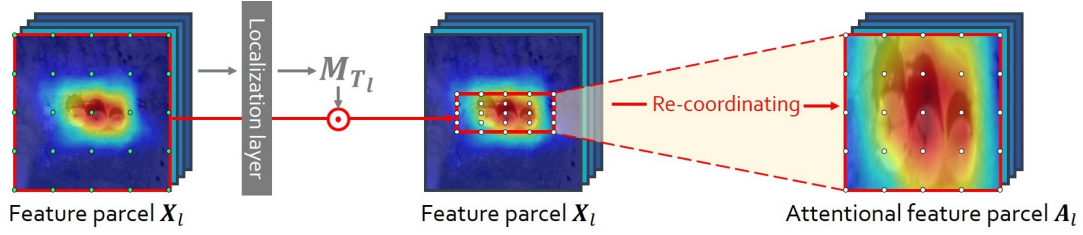


Fig. 3: Illustration of the attentional region extraction module. Green dots in the left image indicate the feature parcel grid G_{X_l} . White dots in the middle image represent the attentional feature parcel grid $G_{X_l^{attn}}$, while those in the right image indicate re-coordinated $G_{X_l^{attn}}$. Notably, the structure of re-coordinated $G_{X_l^{attn}}$ is identical to that of G_{X_l} , and values of pixels located at grid points in re-coordinated $G_{X_l^{attn}}$ are obtained from those in G_{X_l} . For example, the pixel at the left top corner grid point in re-coordinated $G_{X_l^{attn}}$ is assigned with the value of that at the left top corner of G_{X_l} .

included. Accordingly, we discard the last max-pooling layer in VGG-16, leading to a spatial size of 14×14 for outputs. Weights are initialized with pre-trained VGG-16 on ImageNet but updated during the training phase.

C. Attentional Region Extraction Module

Although label-wise feature parcels can be directly applied to exploring label dependencies [34], less informative regions (see blue areas in Fig. 3) may bring noise and further reduce the effectiveness of these feature parcels. As shown in the left image of Fig. 3, weakly activated regions indicate a loose relevance to the corresponding label, while highlighted regions suggest a strong region-label relevance. To diminish the influence of unrelated regions, we employ an attentional region extraction module to automatically extract discriminative regions from label-wise feature parcels.

We localize and re-coordinate attentional regions from X_l with a learnable spatial transformer. Particularly, we sample a feature parcel X_l into a regular spatial grid G_{X_l} (cf. green dots in the left image of Fig. 3) according to the spatial resolution of X_l and regard pixels in X_l as points on the grid G_{X_l} with coordinates (x_l, y_l) . Similarly, we can define coordinates of a new grid, attentional region grid $G_{X_l^{attn}}$ (see white dots in the middle image of Fig. 3), as (x_l^{attn}, y_l^{attn}) , and the number of grid points along with the height and width is equivalent to that of G_{X_l} . As demonstrated in [49] that $G_{X_l^{attn}}$ can be learned by performing spatial transformation on G_{X_l} , (x_l^{attn}, y_l^{attn}) can be calculated with the following equation:

$$\begin{bmatrix} x_l^{attn} \\ y_l^{attn} \end{bmatrix} = M_{T_l} \begin{bmatrix} x_l \\ y_l \\ 1 \end{bmatrix}, \quad (1)$$

where M_{T_l} is a learnable transformation matrix, and grid coordinates, x_l and y_l , are normalized to $[-1, 1]$. Considering that this module is designed for localization, we only adopt scaling and translation in our case. Hence Eq. 1 can be rewritten as

$$\begin{bmatrix} x_l^{attn} \\ y_l^{attn} \end{bmatrix} = \begin{bmatrix} s_{x_l} & 0 & t_{x_l} \\ 0 & s_{y_l} & t_{y_l} \end{bmatrix} \begin{bmatrix} x_l \\ y_l \\ 1 \end{bmatrix}, \quad (2)$$

where s_{x_l} and s_{y_l} indicate scaling factors along x- and y-axis, respectively, and t_{x_l} and t_{y_l} represent how feature maps

should be translated along both axes. Notably, since different objects distribute variously in aerial images, M_{T_l} is learned for each object label l individually. In other words, extracted attentional regions are label-specific and capable of improving the effectiveness of label-wise features.

As to the implementation of this module, we first vectorize X_l with a flatten function and then employ a localization layer (e.g., a fully connected layer) to estimate elements in M_{T_l} from the vectorized X_l . Afterwards, attentional region grid coordinates (x_l^{attn}, y_l^{attn}) can be learned from (x_l, y_l) with Eq. 2, and values of pixels at (x_l^{attn}, y_l^{attn}) is able to be obtained from neighboring pixels by bilinear interpolation. Finally, the attentional region grid $G_{X_l^{attn}}$ is re-coordinated to a regular spatial grid, which shares an identical structure with G_{X_l} , for yielding the final attentional feature parcel A_l .

D. Label Relational Inference Module

Being the core of our model, the label relational inference module is designed to fully exploit label interrelations for inferring existences of all labels. Before diving into this module, we define the pairwise label relation as a composite function with the following equation:

$$\text{LR}(A_l, A_m) = f_\phi(g_{\theta_{lm}}(A_l, A_m)), \quad (3)$$

where the input is a pair of attentional feature parcels, A_l and A_m , and l and m range from 1 to L . The functions $g_{\theta_{lm}}$ and f_ϕ are used to reason about the pairwise relation between label l and m . More specifically, the role of $g_{\theta_{lm}}$ is to reason about whether there exist relations between the two objects and how they are related. In previous works [44], [47], a multilayer perceptron (MLP) is commonly employed as $g_{\theta_{lm}}$ for its simplicity. However, spatial contextual semantics are not taken into account in this way. To address such issue, here, we make use of 1×1 convolution instead of an MLP to explore spatial information. Furthermore, f_ϕ is applied to encode the output of $g_{\theta_{lm}}$ into the final pairwise label relation $\text{LR}(A_l, A_m)$. In our case, f_ϕ consists of a global average pooling layer and an MLP, which finally yields the relation between label l and m .

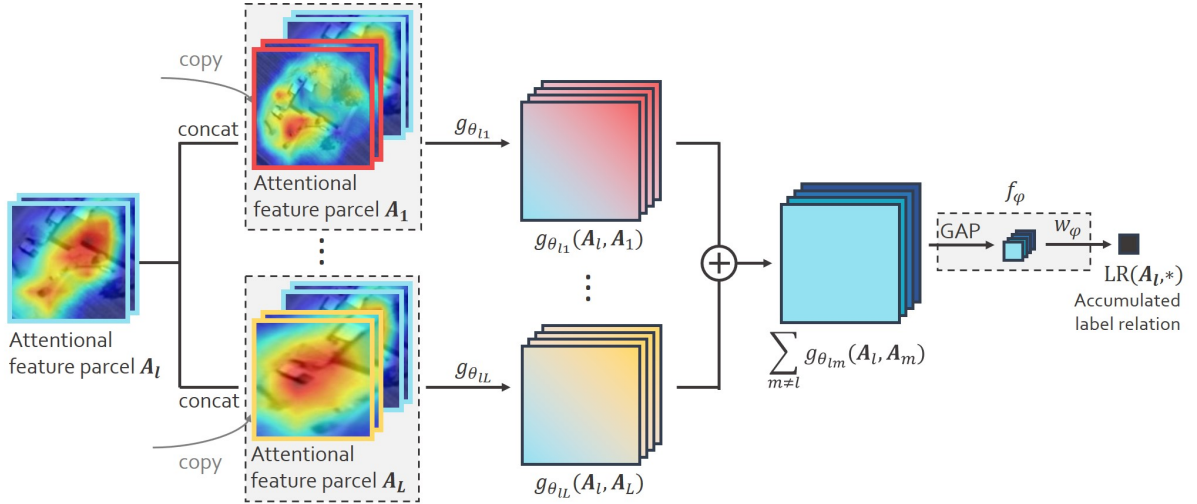


Fig. 4: Illustration of the label relation module.

Following the motivation of our work, we infer each label by accumulating all related pairwise label relations, and the accumulated label relation for object label l is defined as:

$$\text{LR}(\mathbf{A}_l, *) = f_\phi\left(\sum_{m \neq l} g_{\theta_{lm}}(\mathbf{A}_l, \mathbf{A}_m)\right), \quad (4)$$

where $*$ represents all attentional feature parcels except \mathbf{A}_l . Based on this formula, we implement the label relational inference module with the following steps (taking the prediction of label l as an example): 1) \mathbf{A}_l and every other attentional feature parcel are concatenated and fed into a 1×1 convolutional layer, respectively. 2) Afterwards, a global average pooling layer is employed to transform $g_{\theta_{lm}}(\mathbf{A}_l, \mathbf{A}_m)$ into vectors, which are then element-wise added. 3) Finally, the output is fed into an MLP layer with trainable parameters ϕ to produce the accumulated label relation $\text{LR}(\mathbf{A}_l, *)$. Note that $g_{\theta_{lm}}$ is a learnable unit, which models pairwise relations using convolutions. Through the end-to-end training, it could be expected to learn data-driven label relations. Experiments in Section III-D and Section III-E have verified that learned label relations are in line with prior knowledge. Since we expect the model to predict probabilities, an activation function σ is utilized to restrict each output digit to $[0, 1]$. For label l , a digit approaching 1 implies a high probability of its presence, while one closing 0 suggests the absence. Fig. 4 presents an visual illustration of the label relational inference module.

Compared to other multi-label classification methods, our model has three benefits:

- 1) The module can inherently reason about label relations as indicated by Eq. 3 and requires no particular prior knowledge about relations among all objects. That is to say, our network does not need to learn *how to compute label relations* and *which object relations should be considered*. All relations are automatically learned through a data-driven way and proven to meet the reality in our experiments.
- 2) The learning effectiveness is independent of long short-term memory, leading to increased robustness. This

is because, in Eq. 4, accumulated label relations are calculated with a summation function instead of a chain architecture, e.g., an LSTM.

- 3) The function $g_{\theta_{lm}}$ is learned for each object label pair l and m separately, which suggests that pairwise label relations are encoded in a specific way. Besides, our implementation of $g_{\theta_{lm}}$ can extend the applicability of relational reasoning compared to using an MLP.

Since [34] shares the same design philosophy that modeling label relations is crucial, here we emphasize two differences between our network and [34]: 1) the proposed network learns to extract discriminative regions as label-wise features for modeling label relations (cf. Section II-C) instead of directly using entire feature maps as in [34]; 2) the proposed label relation inference module encodes label relations explicitly with composite functions, while in [34], label relations are modeled implicitly via an RNN whose effectiveness depends heavily on the learning effect of long-term memorization. Quantitative comparisons between these two approaches are shown in the following section.

III. EXPERIMENTS AND DISCUSSION

In this section, we conduct experiments on the UCM [50] and proposed AID multi-label dataset for evaluating our model. Specifically, Section III-A presents a description of these two datasets. Afterwards, we introduce training strategies and thoroughly discuss experimental results in the subsequent subsections.

A. Dataset Introduction

1) *UCM multi-label dataset*: UCM multi-label dataset [50] is reproduced by assigning all aerial images collected in UCM dataset [22] with newly defined object labels. The number of all candidate object labels is 17: building, sand, dock, court, tree, sea, bare soil, mobile home, ship, field, tank, water, grass, pavement, chaparral, and car. It is worth noting that labels, such as tank, airplane, and building, exist in both [22] and [50]

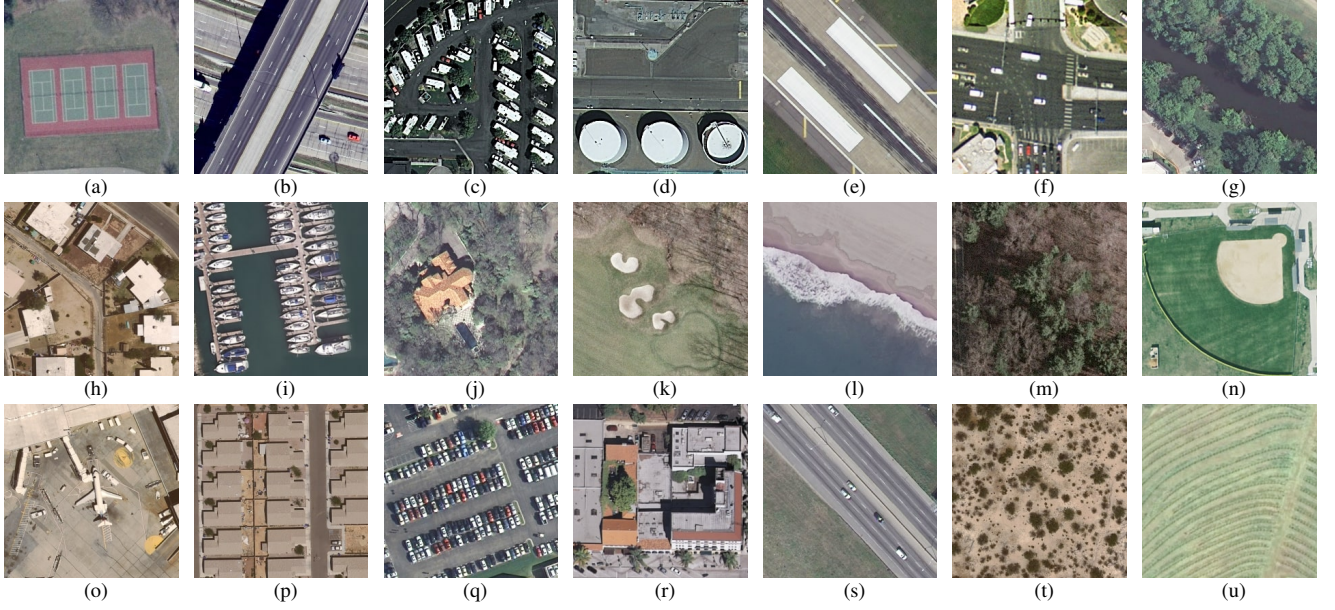


Fig. 5: Samples of various scene categories in the UCM multi-label dataset as well as associated *object* labels. The spatial resolution of each image is one foot, and the size is 256×256 pixels. Scene and *object* labels of each sample are as follows: (a) Tennis court: *tree, grass, court, and bare soil*. (b) Overpass: *pavement, bare soil, and car*. (c) Mobile home park: *pavement, grass, bare soil, tree, mobile home, and car*. (d) Storage tank: *tank, pavement, and bare soil*. (e) Runway: *pavement and grass*. (f) Intersection: *car, tree, pavement, grass, and building*. (g) River: *water, tree, and grass*. (h) Medium residential: *pavement, grass, car, tree, and building*. (i) Harbor: *ship, water, and dock*. (j) Sparse residential: *car, tree, grass, pavement, building, and bare soil*. (k) Golf course: *sand, pavement, tree, and grass*. (l) Beach: *sea and sand*. (m) Forest: *tree, grass, and building*. (n) Baseball diamond: *pavement, grass, building, and bare soil*. (o) Airplane: *airplane, car, bare soil, grass and pavement*. (p) Dense residential: *tree, building, pavement, grass, and car*. (q) Parking lot: *pavement, grass, and car*. (r) building: *pavement, car, and building*. (s) Free way: *tree, car, pavement, grass, and bare soil*. (t) Chaparral: *chaparral and bare soil*. (u) Agricultural: *tree and field*.

while at different levels. In [22], such terms are considered as scene-level labels due to the fact that related images can be characterized and depicted by them, while in [50], they mean objects that may present in aerial images.

As to properties of images in this dataset, the spatial resolution of each sample is one foot, and the size is 256×256 pixels. All images are manually cropped from aerial imagery contributed by the National Map of the U.S. Geological Survey (USGS), and there are 2100 images in total. For each object category, the number of images is listed in Table I. Besides, 80% of image samples per scene class are selected to train our model, and the other 20% of images are used to test our model. Numbers of images assigned to training and test sets with respect to all object labels are available in Table I as well. Some visual examples are shown in Fig. 5.

2) *AID multi-label dataset*: In order to further evaluate our network and meanwhile promote progress in the area of multi-class classification of high-resolution aerial images, we produce a new dataset, named AID multi-label dataset, based on the widely used AID scene classification dataset [23]. The AID dataset consists of 10000 high-resolution aerial images collected from worldwide Google Earth imagery, including scenes from China, the United States, England, France, Italy, Japan, and Germany. In contrast to the UCM dataset, spatial

TABLE I: The number of images for different object categories in the UCM multi-label dataset.

Category No.	Category Name	Training	Test	Total
1	bare soil	577	141	718
2	airplane	80	20	100
3	building	555	136	691
4	car	722	164	886
5	chaparral	82	33	115
6	court	84	21	105
7	dock	80	20	100
8	field	79	25	104
9	grass	804	171	975
10	mobile home	82	20	102
11	pavement	1047	253	1300
12	sand	218	76	294
13	sea	80	20	100
14	ship	80	22	102
15	tank	80	20	100
16	tree	801	208	1009
17	water	161	42	203
-	All	1680	420	2100

resolutions of images in the AID dataset vary from 0.5 m/pixel to 8 m/pixel, and the size of each aerial image is 600×600 pixels. Besides, the number of images in each scene category

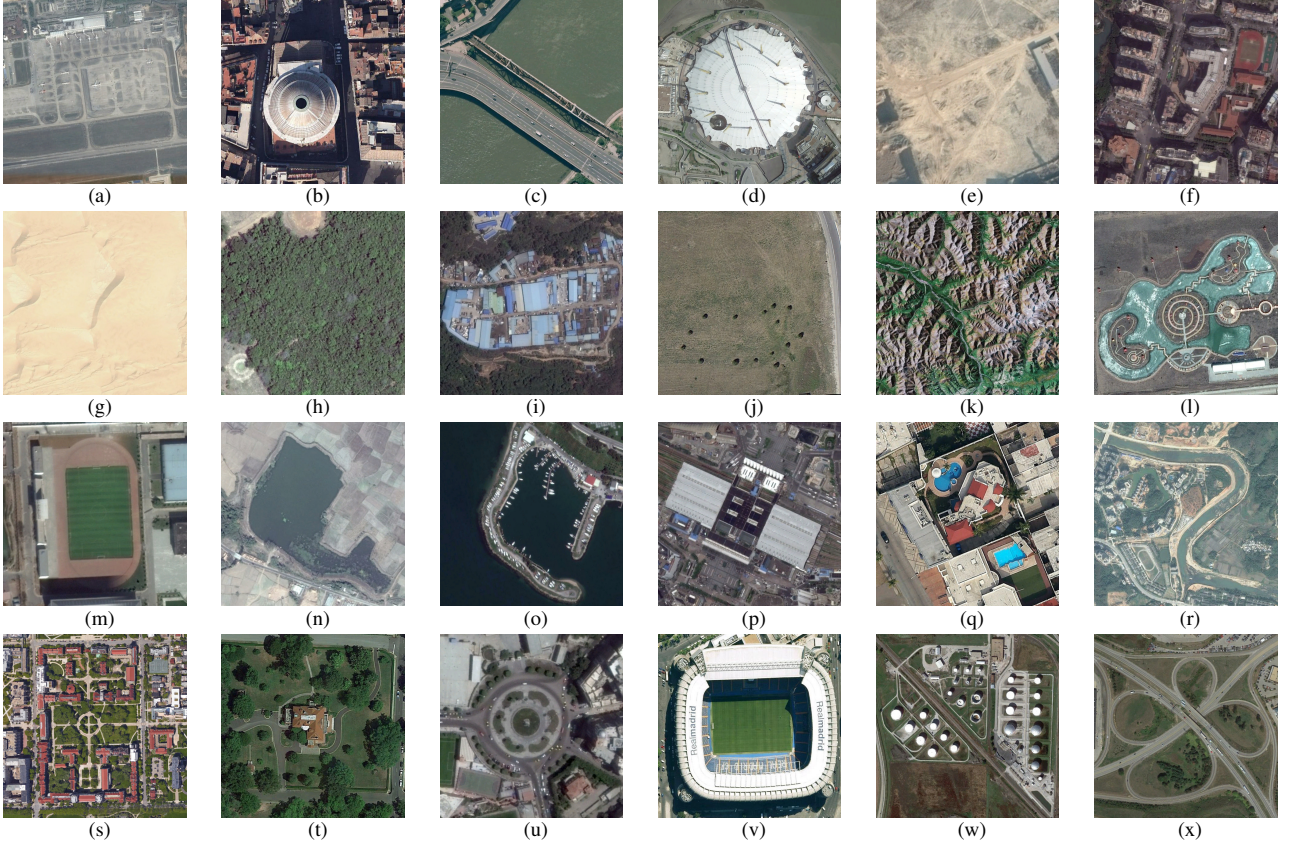


Fig. 6: Samples of various scene categories in the AID multi-label dataset and their associated *object* labels. The spatial resolution of each image varies from 0.5 to 8 m/pixel, and the size is 600×600 pixels. Here are scene and *object* labels of selected samples: (a) Airport: *car, building, tank, tree, airplane, grass, pavement, and bare soil*. (b) Church: *pavement, car, and building*. (c) Bridge: *building, car, grass, pavement, tree and water*. (d) Center: *grass, building, tree, car, bare soil, and pavement*. (e) Bare land: *bare soil, building, pavement, and water*. (f) Commercial: *building, car, court, grass, pavement, tree, and water*. (g) Desert: *sand*. (h) Forest: *bare soil and tree*. (i) Industrial: *pavement, grass, car, bare soil, and building*. (j) Meadow: *pavement and grass*. (k) Mountain: *tree and grass*. (l) Park: *bare soil, building, court, grass, pavement, tree, and water*. (m) Playground: *car, grass, and pavement*. (n) Pond: *building, field, grass, pavement, tree, and water*. (o) Port: *ship, sea, car, grass, pavement, tree, building, and dock*. (p) Railway: *tree, car, pavement, building, and grass*. (q) Resort: *pavement, building, car, tree, field, bare soil, and water*. (r) River: *car, building, bare soil, dock, water, grass, pavement, tree, ship, and field*. (s) School: *pavement, tank, grass, court, building, and car*. (t) Sparse residential: *pavement, car, building, tree, and grass*. (u) Square: *tree, car, court, pavement, grass, and building*. (v) Stadium: *car, pavement, tree, court, grass, building, and bare soil*. (w) Storage tanks: *tank, tree, car, grass, pavement, building, and bare soil*. (x) Viaduct: *pavement, car, bare soil, tree, grass, and building*.

ranges from 220 to 420. Overall, the AID dataset is more challenging compared to the UCM dataset.

Here, we manually relabel some images in the AID dataset. With extensive human visual inspections, 3000 aerial images from 30 scenes in the AID dataset are selected and assigned with multiple object labels, and the distribution of samples in each category is shown in Table II. Besides, 80% of all images are taken as training samples, while the rest is used for testing our model. Several example images are shown in Fig. 6.

B. Training Details

As to the initialization of our network, different modules are done in different ways. For the label-wise feature parcel

learning module, we initialize the backbone and weights in other convolutional layers with a pre-trained ImageNet [57] model and a Glorot uniform initializer, respectively. Regarding the attentional region extraction module, we initialize the transformation matrix in Eq. 1 as an identical transformation,

$$\mathbf{M}_{T_l} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}. \quad (5)$$

In the label relational inference module, weights in both f_ϕ and $g_{\theta_{lm}}$ are initialized with a Glorot uniform initializer and updated during the training phase. Notably, the entire network is trained in an end-to-end manner, and weights in the backbone are fine-tuned as well.

TABLE II: The number of images for different object categories in the AID multi-label dataset.

Category No.	Category Name	Training	Test	Total
1	bare soil	1171	304	1475
2	airplane	79	20	99
3	building	1744	417	2161
4	car	1617	409	2026
5	chaparral	75	37	112
6	court	269	75	344
7	dock	221	50	271
8	field	175	39	214
9	grass	1829	466	2295
10	mobile home	1	1	2
11	pavement	1870	458	2328
12	sand	207	52	259
13	sea	177	44	221
14	ship	237	47	284
15	tank	87	21	108
16	tree	1923	483	2406
17	water	674	178	852
-	All	2400	600	3000

In our case, multiple labels are encoded into multi-hot binary sequences instead of one-hot vectors widely used in single-label classification tasks. The length of such multi-hot binary sequence is identical to the number of total object categories, i.e., 17 in our case, and as to each digit, 0 suggests an absent object, while 1 indicates the presence of its corresponding object label. Accordingly, we define the network loss as the binary cross-entropy. Besides, Adam with Nesterov momentum [58], which shows faster convergence than stochastic gradient descent (SGD) for our task, are selected and its parameters are set as recommended [58]: $\epsilon = 1e - 08$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. The learning rate is initially defined as $1e - 04$ and decayed by a factor of 10 if the validation loss fails to decrease. Notably, we randomly select 10% of the training samples as the validation set. That is, during the training procedure, we use 90% of the training samples to learn network parameters.

Our model is implemented on TensorFlow-1.12.0 and trained for 100 epochs. The computational resource is an NVIDIA Tesla P100 GPU with a 16GB memory. As a compromise between the training speed and GPU memory capacities, we set the size of training batches as 32. To avoid overfitting, the training progress is terminated once the validation loss increases continuously in five epochs.

C. Experimental Setup

To fully explore the capacity of our proposed network, we extend our researches by replacing the backbone with GoogLeNet (Inceptionv3) [59] and ResNet (ResNet-50 in our case) [60]. Specifically, we adapt GoogLeNet by removing global average pooling and fully-connected layers as well as reducing the stride of convolutional and pooling layers in “mixed8” to 1 to improve the spatial resolution. Besides, in order to preserve receptive fields of subsequent convolutional layers, filters in “mixed9” are replaced with atrous convolutional filters, and the dilation rate is defined as 2. Regarding ResNet, we set the convolution stride and dilation rate of filters

as 1 and 2, respectively, in the last residual block. Global average pooling and fully-connected layers are removed as well.

In our experiments, we compare the proposed attention-aware label relational reasoning network (AL-RN-CNN) with the following competitors: a standard CNN, CNN-RBFNN [31], and CA-CNN-BiLSTM [34]. Regarding the CNN, we replace its last softmax layer, designed for single-label classification, with a sigmoid layer to produce multi-hot sequences. For the CA-CNN-BiLSTM, we follow the experimental configurations in [34]. Specifically, we first initialize the feature extraction module of CA-CNN-BiLSTM and weights in the bidirectional LSTM layer with CNNs pre-trained on ImageNet dataset and random values from -0.1 to 0.1, respectively. Afterwards, we fine-tune the entire network in the training phase with Nesterov Adam optimizer, and the initial learning rate is set as $1e - 04$. The loss is calculated with the binary cross-entropy, and the size of training batches is 32. Notably, for all models, output sequences are binarized with a threshold of 0.5 to generate final predictions.

D. Results on the UCM Multi-label Dataset

1) *Quantitative analysis:* In our experiment, we employ F_1 [61] and F_2 [62] scores as evaluation metrics to quantitatively assess the performance of different models. Specifically, these two F scores are calculated with the following equation:

$$F_\beta = (1 + \beta^2) \frac{p_e r_e}{\beta^2 p_e + r_e}, \quad \beta = 1, 2, \quad (6)$$

where p_e indicates the example-based precision and recall [63] of predictions. Formulas of calculating p_e and r_e are:

$$p_e = \frac{TP_e}{TP_e + FP_e}, \quad r_e = \frac{TP_e}{TP_e + FN_e}, \quad (7)$$

where TP_e (example-based true positive) indicates the number of correctly predicted positive labels in an example, while FP_e (example-based false positive) denotes the number of those failed to be recognized. Besides, FN_e (example-based false negative) represents the number of incorrectly predicted negative labels in an example. Here, an example stands for an aerial image and its associated multiple labels.

To evaluate our network comprehensively, we take mean F_1 and F_2 score as principal indexes. Moreover, we also report mean p_e and mean r_e . In addition to the example-based perspective, label-based precision and recall are also considered and calculated with:

$$p_l = \frac{TP_l}{TP_l + FP_l}, \quad r_l = \frac{TP_l}{TP_l + FN_l}, \quad (8)$$

to demonstrate the performance of networks from the perspective of each object label.

Table III exhibits experimental results on the UCM multi-label dataset. We can observe that our model surpasses all competitors on the UCM multi-label dataset with variant backbones. Specifically, AL-RN-VGGNet increases mean F_1 and F_2 scores by 7.16% and 5.64%, respectively, in comparison with VGGNet. Compared to CA-VGG-BiLSTM, which resorts to employing a bidirectional LSTM structure for exploring label dependencies, our network obtains an improvement of

TABLE III: Comparisons of the classification performance on UCM Multi-label Dataset (%).

Network	mean F_1	mean F_2	mean p_e	mean r_e	mean p_l	mean r_l
VGGNet [64]	78.54	80.17	79.06	82.30	86.02	80.21
VGG-RBFNN [31]	78.80	81.14	78.18	83.91	81.90	82.63
CA-VGG-BiLSTM [34]	79.78	81.69	79.33	83.99	85.28	76.52
AL-RN-VGGNet	85.70	85.81	87.62	86.41	91.04	81.71
GoogLeNet [59]	80.68	82.32	80.51	84.27	87.51	80.85
GoogLeNet-RBFNN [31]	81.54	84.05	79.95	86.75	86.19	84.92
CA-GoogLeNet-BiLSTM [34]	81.82	84.41	79.91	87.06	86.29	84.38
AL-RN-GoogLeNet	85.24	85.33	87.18	85.86	91.03	81.64
ResNet-50 [60]	79.68	80.58	80.86	81.95	88.78	78.98
ResNet-RBFNN [31]	80.58	82.47	79.92	84.59	86.21	83.72
CA-ResNet-BiLSTM [34]	81.47	85.27	77.94	89.02	86.12	84.26
AL-RN-ResNet	86.76	86.67	88.81	87.07	92.33	85.95

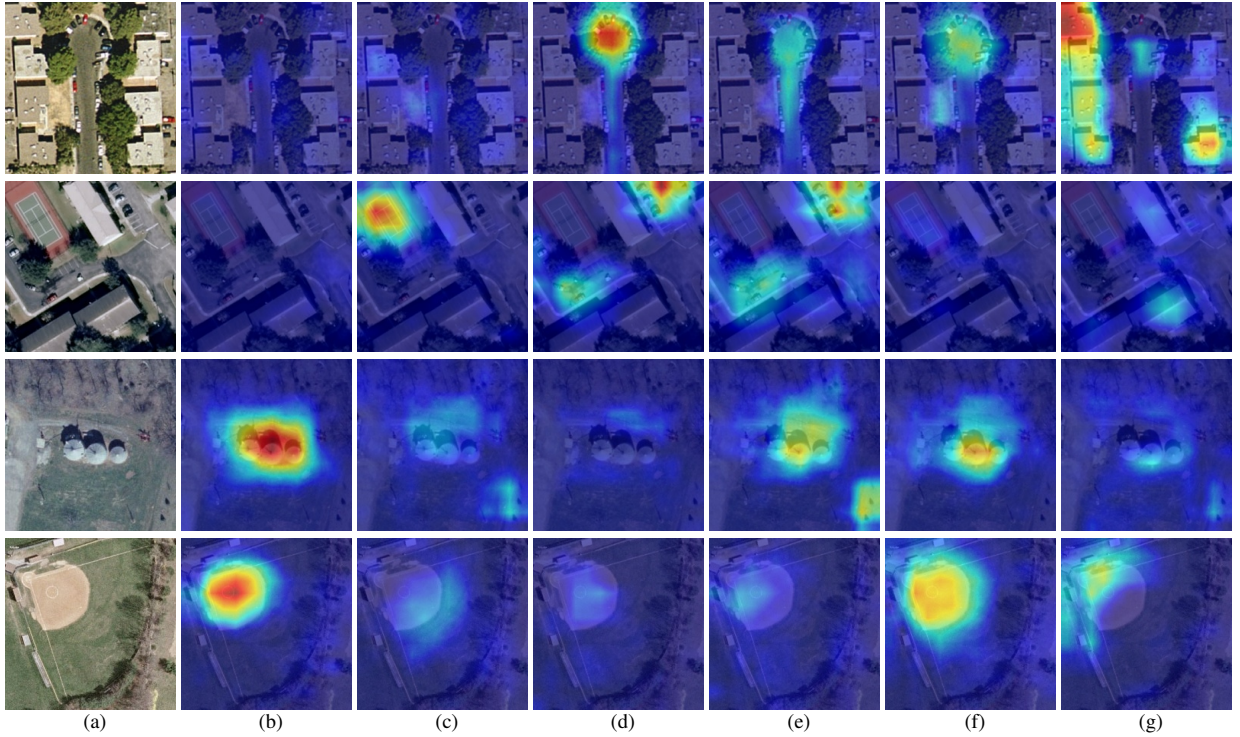


Fig. 7: Example label-specific features of (a) samples selected from the UCM multi-label dataset regarding (b) tank, (c) court, (d) pavement, (e) car, (f) bare soil, and (g) building. Red implies strong activations, while blue indicates weak activations.

5.92% in the mean F_1 score. Besides, although CA-VGG-BiLSTM is superior to VGGNet in both mean F_1 and F_2 scores, it achieves decreased mean precisions and recalls. In contrast, AL-RN-VGGNet outperforms VGGNet not only in mean F_1 and F_2 scores but also in mean example- and label-based precisions and recalls. For another backbone, GoogLeNet, our network gains the best mean F_1 and F_2 scores. As shown in Table III, AL-RN-GoogLeNet increases the mean F_1 score by 4.56% and 3.42% with respect to GoogLeNet and CA-GoogLeNet-BiLSTM, respectively. For the mean F_2 score and precisions, our model also surpasses other competitors, which proves the effectiveness and robustness of our method. AL-RN-ResNet achieves the best mean

F_1 score, 0.8676, and F_2 score, 0.8667, in comparison with all other models. Furthermore, it obtains the best mean example-based precision, 0.8881, and label-based precision, 0.9233, and recall, 0.8595. To summarize, comparisons between AL-RN-CNN and other models demonstrate the effectiveness of our network. Moreover, comparisons between AL-RN-CNN and CA-CNN-BiLSTM illustrate that the composite function-based proposed model performs better than a BiLSTM framework in terms of both accuracy and robustness. Reasons could be that: 1) a chain-like BiLSTM architecture might suffer from the error propagation [41] and thus is sensitive to the order of predictions, while in our network, all pair-wise label relations are encoded separately and the final summation function is

TABLE IV: Example Images and Predicted labels on the UCM and AID Multi-label Dataset.

Samples from the UCM Multi-label Dataset					
Ground Truths	building, car, court, grass, tree, and pavement	building, bare soil, pavement, and grass	car, tree, building, grass, and bare soil	pavement, grass, tree, and bare soil	car, pavement, and building
Predictions	building, car, court, grass, tree, and pavement	building, bare soil, pavement, and grass	tree, car, building, grass, bare soil, and pavement	pavement, grass, tree, and bare soil	car, pavement, and building
Samples from the AID Multi-label Dataset					
Ground Truths	building, car, grass, tree, and pavement	car, bare soil, court, building, grass, tree, pavement, and water	building, car, tree, dock, grass, pavement, sea, and ship	bare soil, building, car, pavement, grass, tree, and water	court, building, car, bare soil, grass, tree, and pavement
Predictions	building, car, grass, tree, and pavement	car, bare soil, court, building, grass, tree, pavement, and water	building, car, tree, dock, grass, pavement, sea, water , and ship	bare soil, car, building, pavement, water, sand , tree, and grass	court, building, car, bare soil, grass, tree, and pavement

Red predictions indicate false positives, while blue predictions are false negatives.

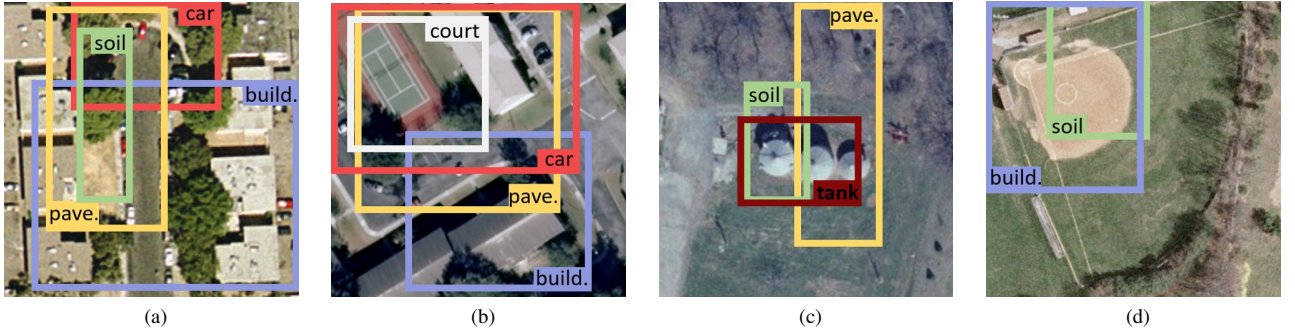


Fig. 8: Example attentional regions for car, bare soil (soil), building (build.), pavement (pave.), court, and tank in various scenes (a)-(d) in the UCM multi-label dataset. For each scene, only positive labels mentioned in Fig. 7 are considered.

order invariant [44]. 2) a BiLSTM-based structure models label relations implicitly, whereas our network encodes such relations in an explicit and direct way. Table IV presents several example predictions from the UCM multi-label dataset. As a supplementary study, we evaluate the robustness of our proposed model by performing cross-validation in the training phase. More specifically, we randomly divide training samples into five folds and train our best-performed model, i.e., AL-RN-ResNet, five times. For each training progress, we select one of five folds as the validation set and train our model with

the remaining four folds. We observe that variances of mean F_1 and F_2 scores are 0.38% and 0.71%, respectively. Compared to improvements brought by our network, variances are limited, and this demonstrates the robustness of our proposed network.

2) *Qualitative analysis*: In order to figure out what is going on inside our network, we further visualize features learned from each module and validate the effectiveness of the proposed network in a qualitative manner. In Fig. 7, a couple of feature parcels regarding bare soil, building, car, pavement,

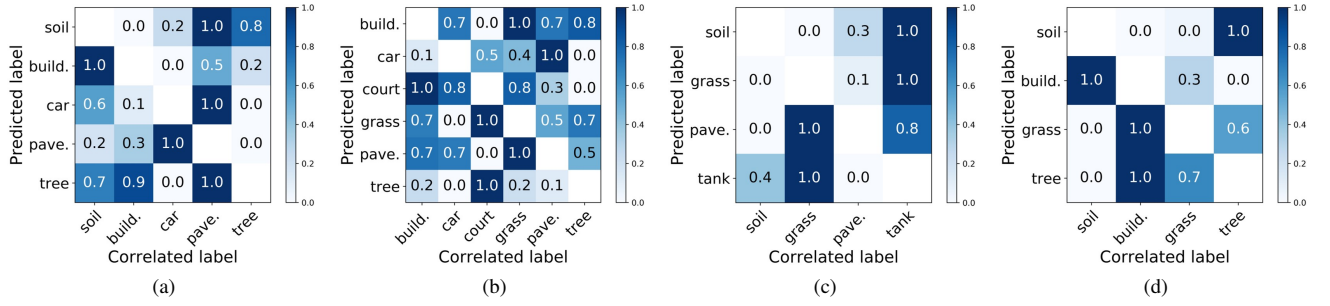


Fig. 9: Example pairwise relations among labels present in scene (a)-(d), which are shown in Fig. 8. Each label at Y-axis represents the predicted label l , and labels at X-axis are correlated labels. Normalization is performed according to each row, and white color represents null values.

TABLE V: Comparisons of the classification performance on AID Multi-label Dataset (%).

Network	mean F_1	mean F_2	mean p_e	mean r_e	mean p_l	mean r_l
VGGNet [64]	85.52	85.60	87.41	86.32	70.60	58.89
VGG-RBFNN [31]	84.58	85.99	84.56	87.85	62.90	69.15
CA-VGG-BiLSTM [34]	86.68	86.88	88.68	87.83	72.04	60.00
proposed AL-RN-VGGNet	88.09	88.31	89.96	89.27	76.94	68.31
GoogLeNet [59]	86.27	85.77	89.49	86.00	74.18	53.69
GoogLeNet-RBFNN [31]	84.85	86.80	84.68	89.14	65.41	72.26
CA-GoogLeNet-BiLSTM [34]	85.36	85.21	88.05	85.79	68.80	59.36
proposed AL-RN-GoogLeNet	88.17	88.25	90.03	88.77	77.92	69.50
ResNet-50 [60]	86.23	85.57	89.31	85.65	72.39	52.82
ResNet-RBFNN [31]	83.77	85.87	82.84	88.32	60.85	70.45
CA-ResNet-BiLSTM [34]	87.63	88.03	89.03	88.99	79.50	65.60
proposed AL-RN-ResNet	88.72	88.54	91.00	88.95	80.81	71.12

court, and tank is displayed for several example images. Note that for K feature maps in each feature parcel, we select the most strongly activated one as the representative. We can observe that discriminative regions related to positive labels are highlighted in these feature maps, while less informative regions are weakly activated. As an exception, the feature map at the bottom left of Fig. 7 shows that the baseball field is misidentified as tanks, which may lead to incorrect predictions.

For evaluating the localization ability of the proposed network, we visualize attentional regions learned from the second module. Coordinates of bottom left (BL) and top right (TR) corners of attentional region grids are calculated with the following equation:

$$\begin{bmatrix} x_{BL}^{attn} & x_{TR}^{attn} \\ y_{BL}^{attn} & y_{TR}^{attn} \end{bmatrix} = \mathbf{M}_{T_i} \begin{bmatrix} -1 & 1 \\ -1 & 1 \\ 1 & 1 \end{bmatrix}. \quad (9)$$

Fig. 8 shows some examples of learned attentional regions. As we can see, most attentional regions concentrate on areas covering objects of interest. Besides, it is noteworthy that even objects are distributed dispersedly, the learned attentional regions can still cover most of them, e.g., buildings in Fig. 8a and cars in 8b.

Furthermore, learned pairwise label relations are visualized in the format of matrix, where an element at (l, m) indicates $LR(\mathbf{A}_l, \mathbf{A}_m)$. Fig. 9 exhibits some examples for the four scenes in Fig. 8. In these examples, we take only positive

object labels into consideration and perform normalization alongside each row to yield a distinct visualization of “*label relations*”. Since m differs from l , we assign null values to diagonal elements and mark them as white color in Fig. 9. It can be seen that in Fig. 9a and 9b, relations between car and pavement contribute significantly to predicting presences of both car and pavement. Besides, Fig. 9d shows that the existence of tree highly suggests the presence of bare soil, but not vice versa. These observations illustrate that even without prior knowledge, the proposed network can reason about relations, that are in line with the reality.

E. Results on the AID Multi-label Dataset

1) *Quantitative analysis*: To further evaluate the proposed network, we report experimental results on the AID multi-label dataset. Evaluation metrics here are the same as those in previous experiments, and results are presented in Table V. As we can observe, the proposed AL-RN-CNN behaves superior to all competitors in most of the metrics. To be more specific, AL-RN-VGGNet improves the mean F_1 and F_2 score by 2.57% and 2.71%, respectively, compared to the baseline model. In comparison with CA-VGG-BiLSTM, our network gains an improvement of 1.41% in the mean F_1 score and 1.43% in the mean F_2 score. Regarding the other two backbones, similar phenomena can be observed as well. AL-RN-GoogLeNet achieves the highest mean F_1 and

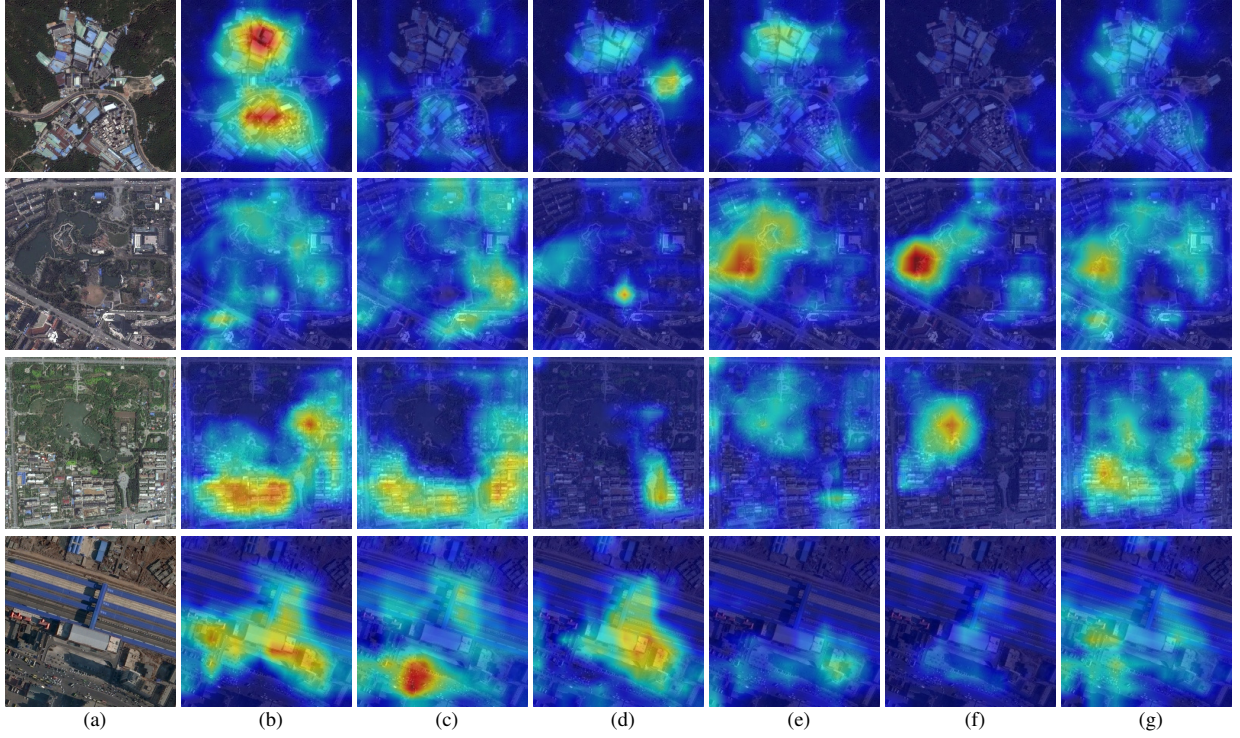


Fig. 10: Example label-specific features of (a) samples selected from the AID multi-label dataset regarding (b) building, (c) car, (d) bare soil, (e) tree, (f) water, and (g) pavement. Red implies strong activations, while blue indicates weak activations.

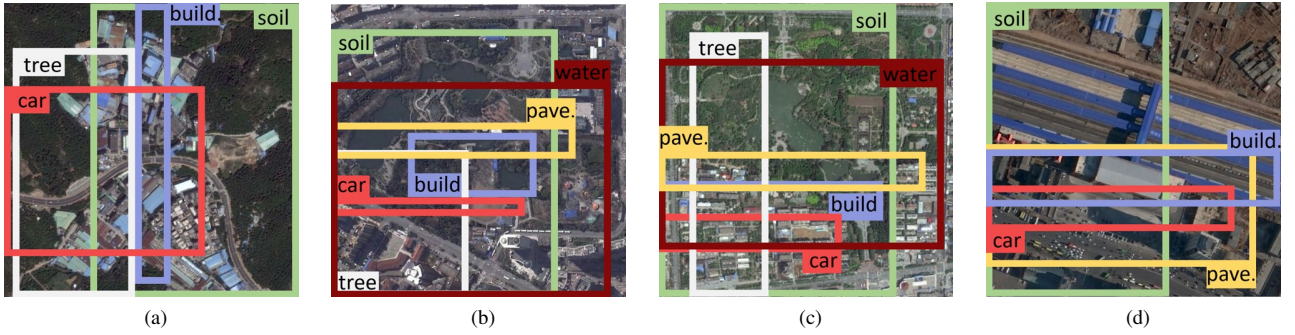


Fig. 11: Example attentional regions for car, bare soil (soil), building (build.), pavement (pave.), court, and tank in various scenes (a)-(d) in the AID multi-label dataset. For each scene, only positive labels mentioned in Fig. 10 are considered.

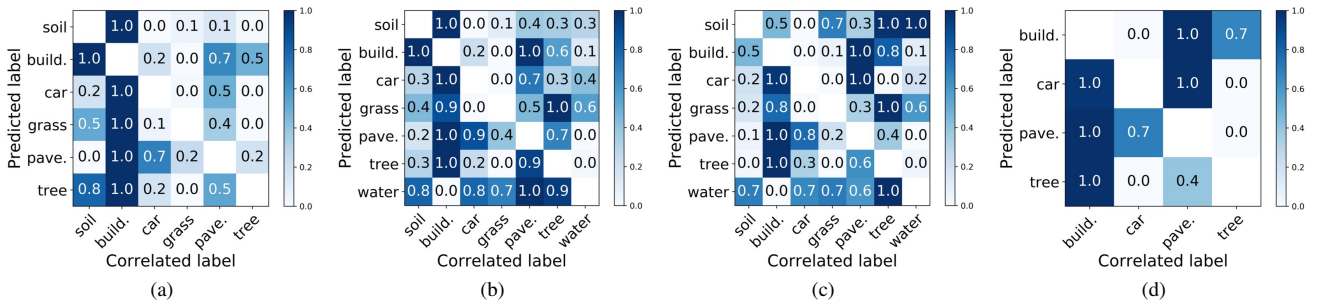


Fig. 12: Example pairwise relations among labels present in scene (a)-(d), which are shown in Fig. 11. Each label at Y-axis represents the predicted label l , and labels at X-axis are correlated labels. Normalization is performed according to each row, and white color represents null values.

TABLE VI: Comparison between different $g_{\theta_{lm}}$ (%).

Dataset	$g_{\theta_{lm}}$	$V*F_1$	$G*F_1$	$R*F_1$	$V*F_2$	$G*F_2$	$R*F_2$
UCM mul.	MLP	82.11	83.02	85.36	81.99	84.02	86.09
	Conv.	85.70	85.24	86.76	85.81	85.33	86.67
AID mul.	MLP	87.79	84.92	87.10	87.74	86.97	86.83
	Conv.	88.09	88.17	88.72	88.31	88.25	88.54

$V*F_1$, $G*F_1$, and $R*F_1$ indicate the mean F_1 score achieved by VGGNet-, GoogLeNet-, and ResNet-based networks.

$V*F_2$, $G*F_2$, and $R*F_2$ indicate the mean F_2 score achieved by VGGNet-, GoogLeNet-, and ResNet-based networks.

F_2 score, 0.8817 and 0.8825, compared to GoogLeNet and CA-GoogLeNet-BiLSTM, while AL-RN-ResNet surpasses the second-best model by 1.09% and 0.51% in the mean F_1 and F_2 score, respectively. Besides, it is noteworthy that although CA-GoogLeNet-BiLSTM shows a decreased performance compared to the baseline model, our network still achieves higher scores in all metrics. Moreover, we notice that the proposed AL-RN-CNNs outperform baseline CNNs by a large margin in the mean label-based recall, and the maximum improvement can reach 18.30%. In conclusion, these comparisons suggest that explicitly modeling label relations can improve the robustness and retrieval ability of a network. Several example predictions on the AID multi-label dataset are presented in Table IV.

2) *Qualitative analysis*: To dive deep into the model, we visualize label-specific features and attentional regions in Fig. 10 and 11, respectively. In Fig. 10, representative feature maps in various feature parcels for bare soil, building, car, pavement, tree, and water are displayed. As shown here, regions with label-related semantics are highlighted, while less informative regions present weak activations. For instance, regions of ponds are considered as discriminative regions for identifying *water*. Residential and industrial areas are strongly activated in feature maps for recognizing *building*. In Fig. 11, it can be observed that attentional regions learned from our network are able to capture areas of semantic objects, such as cars and trees. We also note that some attentional regions in Fig. 11 are coarser than those in Fig. 8, which is because the AID multi-label dataset has a lower spatial resolution.

Furthermore, pairwise relations among positive labels are visualized in Fig. 12. As shown in Fig. 12b, 12c, and 12d, existences of both tree and pavement contribute significantly to the identification of car, while the occurrence of car only suggests a high probability that pavement presents. Strong pairwise relations between building and other labels, e.g., car, pavement, and tree, indicate that the presence of building can heavily assist in predicting those labels.

F. Discussion on the Relational Inference Module

Regarding the relational inference module, the function $g_{\theta_{lm}}$ is an important component, which reasons about relations between two objects. Hence, in this subsection, we discuss about different implementations of $g_{\theta_{lm}}$. Specifically, we compare our AL-RN-CNN with LR-CNN [65], which employs a global average pooling layer and an MLP as $g_{\theta_{lm}}$, on both the UCM and AID multi-label datasets. Experimental

results are reported in Table VI. As shown in this table, our network gains the best mean F_1 and F_2 score on both datasets with variant backbones. AL-RN-VGGNet achieves the highest improvements of 3.59% and 3.82% for the mean F_1 and F_2 score, respectively, compared to LR-VGGNet on the UCM multi-label dataset. AL-RN-GoogLeNet increases the mean F_1 and F_2 score by 3.25% and 1.28%, respectively, in comparison with LR-ResNet on the AID multi-label dataset. Moreover, AL-RN-CNN can encode label relations through various fields of view by simply changing the size of convolutional filters in $g_{\theta_{lm}}$.

IV. CONCLUSION

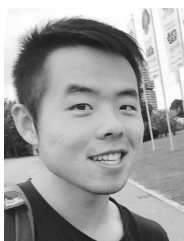
In this work, we propose a novel aerial image multi-label classification network, namely attention-aware label relational reasoning network. This network comprises three components: a label-wise feature parcel learning module, an attentional region extraction module, and a label relational inference module. To be more specific, the label-wise feature parcel learning module is designed to learn high-level feature parcels, which are proven to encompass label-relevant semantics, and the attentional region extraction module further generates finer attentional feature parcels by preserving only features located in discriminative regions. Afterwards, the label relational inference module reasons about pairwise relations among all labels and exploit these relations for the final prediction. In order to assess the performance of our network, experiments are conducted on the UCM multi-label dataset and a newly proposed AID multi-label dataset. In comparison with other deep learning methods, our network can offer better classification results. In addition, we visualize extracted feature parcels, attentional regions, and relation matrices for demonstrating the effectiveness of each module in a qualitative way. Looking into the future, such network architecture has several potentials, e.g., weakly supervised object detection and semantic segmentation.

REFERENCES

- [1] D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Datcu, and U. Stilla, "Classification with an edge: Improving semantic image segmentation with boundary detection," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 135, no. January, pp. 158–172, 2018.
- [2] N. Audebert, B. L. Saux, and S. Lefèvre, "Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 140, no. June, pp. 20–32, 2018.
- [3] D. Marcos, M. Volpi, B. Kellenberger, and D. Tuia, "Land cover mapping at very high resolution with rotation equivariant CNNs: Towards small yet accurate models," *ISPRS Journal of Photogrammetry and Remote Sensing*, DOI:10.1016/j.isprsjprs.2018.01.021.
- [4] L. Mou and X. X. Zhu, "RifCN: Recurrent network in fully convolutional network for semantic segmentation of high resolution remote sensing images," *arXiv:1805.02091*, 2018.
- [5] L. Mou and X. X. Zhu, "Vehicle instance segmentation from aerial image and video using a multi-task learning residual fully convolutional network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 11, pp. 6699–6711, 2018.
- [6] L. Mou and X. X. Zhu, "Spatiotemporal scene interpretation of space videos via deep neural network and tracklet analysis," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2016.
- [7] Q. Li, L. Mou, Q. Liu, Y. Wang, and X. X. Zhu, "Hsf-net: Multi-scale deep feature embedding for ship detection in optical remote sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, 2017.

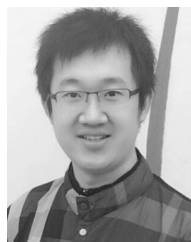
- [8] S. Lucchesi, M. Giardino, and L. Perotti, "Applications of high-resolution images and DTMs for detailed geomorphological analysis of mountain and plain areas of NW Italy," *European Journal of Remote Sensing*, vol. 46, no. 1, pp. 216–233, 2013.
- [9] L. Mou and X. X. Zhu, "IM2HEIGHT: Height estimation from single monocular imagery via fully residual convolutional-deconvolutional network," *arXiv:1802.10249*, 2018.
- [10] Q. Weng, Z. Mao, J. Lin, and X. Liao, "Land-use scene classification based on a CNN using a constrained extreme learning machine," *International Journal of Remote Sensing*, vol. 0, no. 0, pp. 1–19, 2018.
- [11] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1865–1883, 2017.
- [12] P. Zarco-Tejada, R. Diaz-Varela, V. Angileri, and P. Loudjani, "Tree height quantification using very high resolution imagery acquired from an unmanned aerial vehicle (UAV) and automatic 3D photo-reconstruction methods," *European Journal of Agronomy*, vol. 55, pp. 89–99, 2014.
- [13] D. Wen, X. Huang, H. Liu, W. Liao, and L. Zhang, "Semantic classification of urban trees using very high resolution satellite imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 4, pp. 1413–1424, 2017.
- [14] K. Nogueira, O. Penatti, and J. dos Santos, "Towards better exploiting convolutional neural networks for remote sensing scene classification," *Pattern Recognition*, vol. 61, pp. 539–556, 2017.
- [15] X. X. Zhu, D. Tuia, L. Mou, G. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geoscience and Remote Sensing Magazine*, vol. 5, no. 4, pp. 8–36, 2017.
- [16] B. Demir and L. Bruzzone, "Histogram-based attribute profiles for classification of very high resolution remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 4, pp. 2096–2107, 2016.
- [17] F. Hu, G. Xia, J. Hu, and L. Zhang, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," *Remote Sensing*, vol. 7, no. 11, pp. 14 680–14 707, 2015.
- [18] F. Hu, G. Xia, Y. W., and Z. L., "Recent advances and opportunities in scene classification of aerial images with deep models," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2018.
- [19] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 4, pp. 2175–2184, 2015.
- [20] X. Huang, H. Chen, and J. Gong, "Angular difference feature extraction for urban scene classification using ZY-3 multi-angle high-resolution satellite imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 135, pp. 127 – 141, 2018.
- [21] L. Mou, X. Zhu, M. Vakalopoulou, K. Karantzalos, N. Paragios, B. L. Saux, G. Moser, and D. Tuia, "Multitemporal very high resolution from space: Outcome of the 2016 IEEE GRSS data fusion contest," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 8, pp. 3435–3447, 2017.
- [22] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*, 2010.
- [23] G. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, and X. Lu, "Aid: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, 2017.
- [24] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015.
- [25] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [26] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *arXiv:1511.00561*, 2015.
- [27] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001.
- [28] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *IEEE conference on computer vision and pattern recognition (CVPR)*, 2017.
- [29] S. Ren, K. He, R. Girshick, X. Zhang, and J. Sun, "Object detection networks on convolutional feature maps," *IEEE transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 7, pp. 1476–1481, 2017.
- [30] K. Karalas, G. Tsagkatakis, M. Zervakis, and P. Tsakalides, "Land classification using remotely sensed data: Going multilabel," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 6, pp. 3548–3563, 2016.
- [31] A. Zeggada, F. Melgani, and Y. Bazi, "A deep learning approach to UAV image multilabeling," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 5, pp. 694–698, 2017.
- [32] S. Koda, A. Zeggada, F. Melgani, and R. Nishii, "Spatial and structured SVM for multilabel image classification," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–13, 2018.
- [33] A. Zeggada, S. Benbraika, F. Melgani, and Z. Mokhtari, "Multilabel conditional random field classification for UAV images," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 3, pp. 399–403, 2018.
- [34] Y. Hua, L. Mou, and X. X. Zhu, "Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional LSTM network for multilabel aerial image classification," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 149, pp. 188–199, 2019.
- [35] W. Shao, W. Yang, G. Xia, and G. Liu, "A hierarchical scheme of multiple feature fusion for high-resolution satellite scene categorization," in *International Conference on Computer Vision Systems*, 2013.
- [36] V. Risojevic and Z. Babic, "Fusion of global and local descriptors for remote sensing image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 4, pp. 836–840, 2013.
- [37] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [38] Q. Zhu, Y. Zhong, B. Zhao, G. S. Xia, and L. Zhang, "Bag-of-visual-words scene classifier with local and global features for high spatial resolution remote sensing imagery," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 6, pp. 747–751, 2016.
- [39] B. Teshome Zegeye and B. Demir, "A novel active learning technique for multi-label remote sensing image scene classification," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, 2018.
- [40] R. Stivaktakis, G. Tsagkatakis, and P. Tsakalides, "Deep learning for multilabel land cover scene categorization using data augmentation," *IEEE Geoscience and Remote Sensing Letters*, 2019.
- [41] G. Sumbul and D. B., "A CNN-RNN framework with a novel patch-based multi-attention mechanism for multi-label image classification in remote sensing," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019.
- [42] G. Sumbul, R. Cinbis, and S. Aksoy, "Fine-grained object recognition and zero-shot learning in remote sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 2, pp. 770–779, 2017.
- [43] C. Lee, C. Yeh, and Y. F. Wang, "Multi-label zero-shot learning with structured knowledge graphs," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [44] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap, "A simple neural network module for relational reasoning," in *Advances in Neural Information Processing Systems (NIPS)*, 2017.
- [45] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei, "Relation networks for object detection," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [46] L. Wang, W. Li, W. Li, and L. Van Gool, "Appearance-and-relation networks for video classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [47] B. Zhou, A. Andonian, and A. Torralba, "Temporal relational reasoning in videos," in *European Conference on Computer Vision (ECCV)*, 2018.
- [48] L. Mou, Y. Hua, and X. X. Zhu, "Relation matters: Relational context-aware fully convolutional network for semantic segmentation of high resolution aerial images," *arXiv:1409.1556*, 2019.
- [49] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2015.
- [50] B. Chaudhuri, B. Demir, S. Chaudhuri, and L. Bruzzone, "Multilabel remote sensing image retrieval using a semisupervised graph-theoretic method," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 2, pp. 1144–1158, 2018.
- [51] F. Hu, G. Xia, W. Yang, and L. Zhang, "Mining deep semantic representations for scene classification of high-resolution remote sensing imagery," *IEEE Transactions on Big Data*, pp. 1–1, 2019.

- [52] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, no. 2, pp. 22–40, 2016.
- [53] S. Srivastava, J. E. Vargas-Muñoz, and D. Tuia, "Understanding urban landuse from the above and ground perspectives: A deep learning, multimodal solution," *Remote Sensing of Environment*, vol. 228, pp. 129–143, 2019.
- [54] W. Huang, Q. Wang, and X. Li, "Feature sparsity in convolutional neural networks for scene classification of remote sensing image," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019.
- [55] C. Qiu, L. Mou, M. Schmitt, and X. X. Zhu, "Local climate zone-based urban land cover classification from multi-seasonal Sentinel-2 images with a recurrent residual network," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 154, pp. 151–162, 2019.
- [56] C. Qiu, L. Mou, M. Schmitt, and X. X. Zhu, "Fusing multi-seasonal Sentinel-2 imagery for urban land cover classification with residual convolutional neural networks," *IEEE Geoscience and Remote Sensing Letters*, 2019, in press.
- [57] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
- [58] T. Dozat, "Incorporating Nesterov momentum into Adam," http://cs229.stanford.edu/proj2015/054_report.pdf, online.
- [59] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [60] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [61] X. Wu and Z. Zhou, "A unified view of multi-label performance measures," *arXiv:1609.00288*, 2016.
- [62] "Planet: Understanding the Amazon from space," <https://www.kaggle.com/c/planet-understanding-the-amazon-from-space#evaluation>, online.
- [63] G. Tsoumakas and I. Vlahavas, "Random k-labelsets: An ensemble method for multilabel classification," in *European Conference on Machine Learning*, 2007.
- [64] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv:1409.1556*, 2014.
- [65] Y. Hua, L. Mou, and X. X. Zhu, "Label relation inference for multi-label aerial image classification," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019.



Yuansheng Hua (S'18) received the bachelor's degree in remote sensing science and technology from the Wuhan University, Wuhan, China, in 2014, and the master's degree in Earth Oriented Space Science and Technology (ESPACE) from the Technical University of Munich (TUM), Munich, Germany, in 2018. He is currently pursuing the Ph.D. degree with the German Aerospace Center (DLR), Wessling, Germany and the Technical University of Munich (TUM), Munich, Germany.

In 2019, he was a visiting researcher with the Wageningen University & Research, Wageningen, Netherlands. His research interests include remote sensing, computer vision, and deep learning, especially their applications in remote sensing.



Lichao Mou (S'16) received the bachelor's degree in automation from the Xi'an University of Posts and Telecommunications, Xi'an, China, in 2012, and the master's degree in signal and information processing from the University of Chinese Academy of Sciences (UCAS), Beijing, China, in 2015. He is currently pursuing the Ph.D. degree with the German Aerospace Center (DLR), Wessling, Germany, and also with the Technical University of Munich (TUM), Munich, Germany. In 2015, he spent six months at the Computer Vision Group, University of Freiburg, Freiburg im Breisgau, Germany.

In 2019, he was a Visiting Researcher with the University of Cambridge, Cambridge, U.K. His research interests include remote sensing, computer vision, and machine learning, especially deep networks and their applications in remote sensing.

Mr. Mou was a recipient of the first place in the 2016 IEEE GRSS Data Fusion Contest and finalists for the Best Student Paper Award at the 2017 Joint Urban Remote Sensing Event and the 2019 Joint Urban Remote Sensing Event.



Xiao Xiang Zhu (S'10–M'12–SM'14) received the Master (M.Sc.) degree, her doctor of engineering (Dr.-Ing.) degree and her "Habilitation" in the field of signal processing from Technical University of Munich (TUM), Munich, Germany, in 2008, 2011 and 2013, respectively.

She is currently the Professor for Signal Processing in Earth Observation (www.sipeo.bgu.tum.de) at Technical University of Munich (TUM) and German Aerospace Center (DLR); the head of the department "EO Data Science" at DLR's Earth Observation Center; and the head of the Helmholtz Young Investigator Group "SiPEO" at DLR and TUM. Since 2019, Zhu is co-coordinating the Munich Data Science Research School (www.mu-ds.de). She is also leading the Helmholtz Artificial Intelligence Cooperation Unit (HAICU) – Research Field "Aeronautics, Space and Transport". Prof. Zhu was a guest scientist or visiting professor at the Italian National Research Council (CNR-IREA), Naples, Italy, Fudan University, Shanghai, China, the University of Tokyo, Tokyo, Japan and University of California, Los Angeles, United States in 2009, 2014, 2015 and 2016, respectively. Her main research interests are remote sensing and Earth observation, signal processing, machine learning and data science, with a special application focus on global urban mapping.

Dr. Zhu is a member of young academy (Junge Akademie/Junges Kolleg) at the Berlin-Brandenburg Academy of Sciences and Humanities and the German National Academy of Sciences Leopoldina and the Bavarian Academy of Sciences and Humanities. She is an associate Editor of IEEE Transactions on Geoscience and Remote Sensing.

C Yuansheng Hua, Lichao Mou, Jianzhe Lin, Konrad Heidler, and Xiao Xiang Zhu, “Aerial scene understanding in the wild: Multi-scene recognition via prototype-based memory networks,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 177, pp. 89-102, 2021.

<https://doi.org/10.1016/j.isprsjprs.2021.04.006>

Aerial Scene Understanding in The Wild: Multi-Scene Recognition via Prototype-based Memory Networks

Yuansheng Hua^{a,b}, Lichao Mou^{a,b}, Jianzhe Lin^c, Konrad Heidler^{a,b}, Xiao Xiang Zhu^{a,b,*}

^a*Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Oberpfaffenhofen, 82234 Wessling, Germany*

^b*Data Science in Earth Observation (SiPEO), Technical University of Munich (TUM), Arcisstr. 21, 80333 Munich, Germany*

^c*Electrical and Computer Engineering (ECE), University of British Columbia (UBC), V6T 1Z2, Canada*

Abstract

Aerial scene recognition is a fundamental visual task and has attracted an increasing research interest in the last few years. Most of current researches mainly deploy efforts to categorize an aerial image into one scene-level label, while in real-world scenarios, there often exist multiple scenes in a single image. Therefore, in this paper, we propose to take a step forward to a more practical and challenging task, namely multi-scene recognition in single images. Moreover, we note that manually yielding annotations for such a task is extraordinarily time- and labor-consuming. To address this, we propose a prototype-based memory network to recognize multiple scenes in a single image by leveraging massive well-annotated single-scene images. The proposed network consists of three key components: 1) a prototype learning module, 2) a prototype-inhabiting external memory, and 3) a multi-head attention-based memory retrieval module. To be more specific, we first learn the prototype representation of each aerial scene from single-scene aerial image datasets and store it in an external memory. Afterwards, a multi-head attention-based memory retrieval module is devised to retrieve scene prototypes relevant to query multi-scene images for final predictions. Notably,

*Corresponding author

Email addresses: yuansheng.hua@dlr.de (Yuansheng Hua), lichao.mou@dlr.de (Lichao Mou), jianzhelin@ece.ubc.ca (Jianzhe Lin), konrad.heidler@dlr.de (Konrad Heidler), xiaoxiang.zhu@dlr.de (Xiao Xiang Zhu)

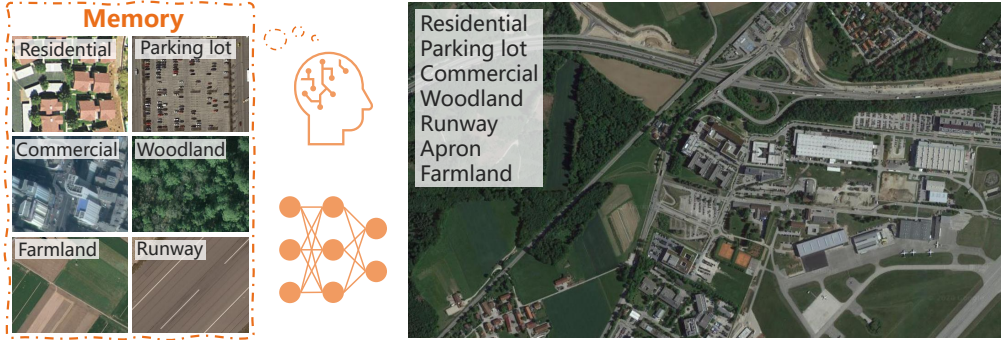


Figure 1: Illustration of how humans learn to perceive unconstrained aerial images being composed of multiple scenes. We first learn and memorize individual aerial scenes. Then we can possess the capability of understanding complex scenarios by learning from only a limited number of hard instances. We believe by simulating this learning process, a deep neural network can also learn to interpret multi-scene aerial images.

only a limited number of annotated multi-scene images are needed in the training phase. To facilitate the progress of aerial scene recognition, we produce a new multi-scene aerial image (MAI) dataset. Experimental results on variant dataset configurations demonstrate the effectiveness of our network. Our dataset and codes are publicly available¹.

Keywords: Convolutional neural network (CNN), multi-scene recognition in single images, memory network, multi-scene aerial image dataset, multi-head attention-based memory retrieval, prototype learning.

1. Introduction

With the enormous advancement of remote sensing technologies, massive high-resolution aerial images are now available and beneficial to a large variety of applications, e.g., urban planning [1, 2, 3, 4, 5, 6, 7], traffic monitoring [8, 9], disaster assessment [10, 11], and natural resource management [12, 13, 14, 15, 16, 17, 18]. Driven by these applications, aerial scene recognition that refers to assigning aerial images scene-level labels is now becoming a fundamental but challenging task.

¹<https://github.com/Hua-YS/Prototype-based-Memory-Network>

In recent years, many efforts [19], e.g., developing novel network architectures [20, 21, 22, 23, 24, 25] and pipelines [26, 27, 28, 29], publishing large-scale datasets [30, 31], introducing multi-modal and multi-temporal data [32, 33, 34, 35], have been deployed to address this task, and most of them treat it as a single-label classification problem. A common assumption shared by these researches is that an aerial image belongs to only one scene category, while in real-world scenarios, it is more often that there exist various scenes in a single image (cf. Figure 1). Furthermore, we notice that aerial images used to learn single-label scene classification models are usually well-cropped so that target scenes could be centered and account for the majority of an aerial image. Unfortunately, this might be infeasible for practical applications. Therefore, in this paper, we aim to deal with a more practical and challenging problem, multi-scene classification in a single image, which refers to inferring multiple scene-level labels for a large-scale, unconstrained aerial image. Figure 1 shows an example image, where we can see that multiple scenes, e.g., **residential**, **parking lot**, and **commercial**, co-exist in one aerial image. We note that there is another research branch of aerial image understanding, multi-label object classification, which refers to the process of inferring multiple objects present in an aerial image. These studies [36, 37, 38, 39, 40, 41, 42] mainly focus on recognizing object-level labels, while in our task, an image is classified into multiple scene categories, which provides a more comprehensive understanding of large-scale aerial images in scene-level. To the best of our knowledge, multi-scene recognition in unconstrained aerial images still remains underexplored in the remote sensing community.

To achieve this task, huge quantities of well-annotated multi-scene images are needed for the purpose of training models. However, we note that such annotations are not easy in the remote sensing community. This could be attributed to the following two reasons. On the one hand, the visual interpretation of multiple scenes is more arduous than that of a single scene in an aerial image, and therefore, labeling multi-scene images requires more work. On the other hand, low-cost annotation techniques, e.g., resorting to crowdsourcing OpenStreetMap (OSM) through keyword searching [30, 31, 43], perform poorly in yielding multi-scene datasets owing to the incompleteness and incorrectness of certain OSM data. Examples of erroneous OSM data are shown in Figure 2. In addition, manually rectifying annotations generated from crowdsourcing data are inevitable due to error-proneness. Such a procedure is quite labor-consuming, as every scene is required to be checked

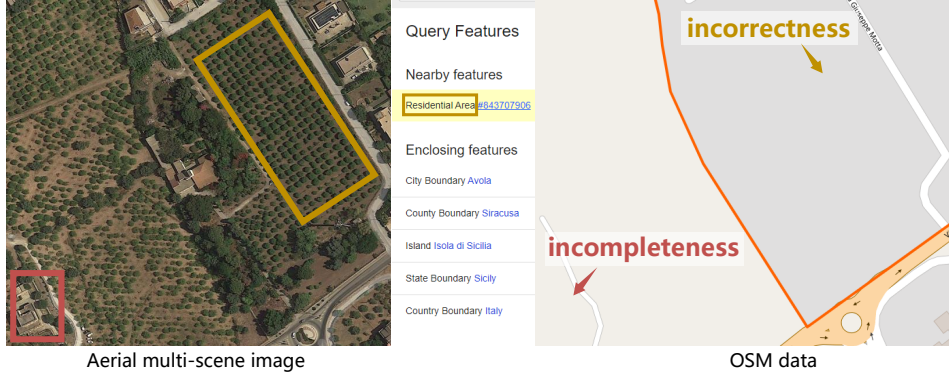


Figure 2: Examples of incomplete (red) and incorrect (yellow) OSM data. Red: the commercial is not annotated in OSM data. Yellow: the orchard is mislabeled as **residential**.

in case that present ones are mislabeled as absent. Aiming to solve the aforementioned limitations, in this work, we propose to train a network for recognizing complex multi-scene aerial images by using only a small number of labeled multi-scene images but a huge amount of existing, annotated single-scene data. Our motivation is based on an intuitive observation about how humans learn to perceive complex scenes being composed of multiple entities [44, 45, 46]: we first learn and memorize individual objects (through flash cards for example) when we were babies and then possess the capability of understanding complex scenarios by learning from only a limited number of hard instances (cf. Figure 1). We believe that this learning process also applies to the interpretation of multi-scene aerial images. Driven by this observation, we propose a novel network, termed as prototype-based memory network (PM-Net), which is inspired by recent successes of memory networks in natural language processing (NLP) tasks [47, 48] and video analysis [49, 50, 51]. To be more specific, we first learn the prototype representation of each aerial scene from single-scene aerial images and then store these prototypes in the external memory of PM-Net. Afterwards, for a given query multi-scene image, a multi-head attention-based memory retrieval module is devised to retrieve scene prototypes that are associated with the query image from the external memory for inferring multiple scene labels.

The contributions of this work are fourfold.

- We take a step forward to a more practical and challenging task in aerial scene understanding, namely multi-scene classification in single

images, which aims to recognize multiple scenes present in a large-scale, unconstrained aerial image. Such a task is in line with real-world scenarios and capable of providing a comprehensive picture for a given geographic area.

- Given that labeling multi-scene images is very labor-intensive and time-consuming, we propose a PM-Net that can be trained for our task by leveraging large numbers of existing single-scene aerial images and a small number of labeled multi-scene images.
- In order to facilitate the progress of multi-scene recognition in single aerial images, we create a new dataset, multi-scene aerial image (MAI) dataset. To the best of our knowledge, this is the first publicly available dataset for aerial multi-scene interpretation. Compared to existing single-scene aerial image datasets, images in our dataset are unconstrained and contain multiple scenes, which are more in line with the reality.
- We carry out extensive experiments with different configurations. Experimental results demonstrate the effectiveness of the proposed network.

The remaining sections of this paper are organized as follows. Section 2 reviews studies in memory networks and prototypical networks, and the architecture of the proposed prototype-based memory network is introduced in Section 3. Section 4 describes experimental configurations and analyzes results. Eventually, conclusions are drawn in Section 5.

2. Related Work

Since very few efforts have been deployed to this task in the remote sensing community, we only review literatures related to our algorithm in this section.

2.1. Memory Networks

A memory network takes as input a query and retrieves complementary information from the external memory. In [47], the memory network is first proposed and utilized to address question-answering tasks, where questions are regarded as queries, and statements are stored in the external memory. To retrieve statements for predicting answers, the authors compute relative

distances between queries and the external memory through dot product. In the following work, Miller et al. [48] improves the efficiency of retrieving large memories by pre-selecting small subsets with key hashing. Moreover, the memory network is further applied in video analysis [49, 50, 51] and image captioning [52]. In [49], the authors devise a dual augmented memory network to memorize both target and background features of an video, and use a Long Short-Term Memory (LSTM) to communicate with previous and next frames. In [50], the authors propose a memory network to memorize normal patterns for detecting anomalies in an video. As an attempt in image captioning, Cornia et al. [52] devise a learnable memory to learn and memorize priori knowledge for encoding relationships between image regions. Inspired by these works, we devise a memory network and store scene prototypes in the memory for recognizing scenes present in multi-scene images.

2.2. Prototypical Networks

Prototypical networks are characterized by classifying images according to their distances from class prototypes. In learning with limited training samples, such networks are popular and achieved many successes recently [53, 54, 55, 56, 57, 58]. To be specific, Snell et al. [53] propose to first learn a prototype representation for each category and then identify images by finding their nearest category prototypes. Guerriero et al. [54] aim to alleviate the heavy expense of learning prototypes by initializing and updating prototypes with those learned in previous training epochs. Yang et al. [55] propose to combine prototypical networks and CNNs for tackling the open world recognition problem and improving the robustness and accuracy of networks. Similarly, Huang et al. [56] propose to integrate prototypical networks and graph convolutional neural networks for learning relational prototypes. Albeit variant, most existing works share a common way to extract prototypes, which is taking average of samples belonging to the same categories. Therefore, we follow this prototype extraction strategy in our work.

3. Methodology

3.1. Overview

The proposed PM-Net consists of three essential components: a prototype learning module, an external memory, and a memory retrieval module. Specifically, the prototype learning module is devised to encode prototype

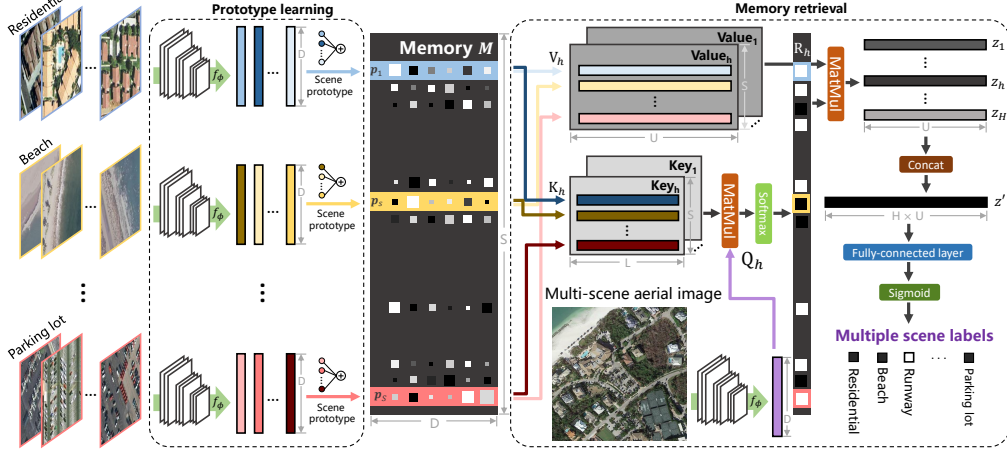


Figure 3: Architecture of the proposed PM-Net. Particularly, we first learn scene prototypes \mathbf{p}_s from well-annotated single-scene aerial images and then store them in the external memory \mathbf{M} of PM-Net. Afterwards, given a query multi-scene image, a multi-head attention-based memory retrieval module is devised to retrieve scene prototypes that are relevant to the query image, yielding \mathbf{z}' for the prediction of multiple labels. f_ϕ denotes the embedding function, and its output is a D -dimensional feature vector. S and H represent numbers of scenes and heads, respectively. L and U denote channel dimensions of the key and value in the memory retrieval module.

representations of aerial scenes, which are then stored in the external memory. The memory retrieval module is responsible for retrieving scene prototypes related to query images through a multi-head attention mechanism. Eventually, retrieved scene prototypes are utilized to infer the existence of multiple scenes in the query image.

3.2. Scene Prototype Learning and Writing

Following the observation introduced in Section 1, we propose to learn and memorize scene prototypes with the support of single-scene aerial images. The procedure consists of two stages. We first employ an embedding function to learn semantic representations of all single-scene images. Then, feature representations belonging to the same scene category are encoded into a scene prototype and stored in the external memory.

Formally, let \mathbf{X}_i^s denote the i -th single-scene image belonging to scene s , and i ranges from 1 to N_s . N_s is the number of samples annotated as s . The

embedding function f_ϕ can be learned via the following objective function:

$$\mathcal{L}(\mathbf{X}_i^s, \mathbf{y}^s) = -\mathbf{y}^s \log \frac{\exp(-g_\theta(f_\phi(\mathbf{X}_i^s)))}{\sum_s \sum_i \exp(-g_\theta(f_\phi(\mathbf{X}_i^s)))}, \quad (1)$$

where ϕ represents learnable parameters of f_ϕ , and \mathbf{y}^s is a one-hot vector denoting the scene label of \mathbf{X}_i^s . g_θ is a multilayer perceptron (MLP) with parameters θ and its outputs are activated by a softmax function to predict probability distributions. Following the overwhelming trend of deep learning, here we employ a deep CNN, e.g., ResNet-50 [59], as the embedding function f_ϕ and learn its parameters on public single-scene aerial image datasets. After sufficient training, f_ϕ is expected to be capable of learning discriminative representations for different aerial scenes.

Once f_ϕ is learned, the scene prototype can be computed by averaging representations of all aerial images belonging to the same scene [53, 54, 55]. Let \mathbf{p}_s be the prototype representation of scene s . We calculate \mathbf{p}_s with the following equation:

$$\mathbf{p}_s = \frac{1}{N_s} \sum_{i=1}^{N_s} f_\phi(\mathbf{X}_i^s). \quad (2)$$

By doing so, in the single-scene classification, an image closely around \mathbf{p}_s in the common embedding space is supposed to belong to scene s . Similarly, in the multi-scene scenario, the representation of an aerial image comprising scene s should show high relevance with \mathbf{p}_s . After encoding all scene prototypes, the external memory \mathbf{M} can be formulated as follows:

$$\mathbf{M} = [\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_S]^T, \quad (3)$$

where S denotes the number of scenes. $[\dots, \dots]$ represents the concatenation operation. Given that \mathbf{p}_s is a D -dimensional vector, \mathbf{M} is a matrix of $S \times D$. Note that D varies when using different backbone CNNs as embedding functions.

3.3. Multi-head Attention-based Memory Retrieval

Inspired by successes of the multi-head self-attention mechanism [60] in natural language processing tasks [61, 62, 63, 64], we develop a multi-head attention-based memory retrieval module to retrieve scene prototypes from the memory \mathbf{M} for a given query image \mathbf{X} . Given a query multi-scene aerial

image \mathbf{X} , to retrieve relevant scene prototypes from \mathbf{M} , we develop a multi-head attention-based memory retrieval module. In particular, we first extract the feature representation of \mathbf{X} through the same embedding function f_ϕ and linearly project it to an L -dimensional query $Q(\mathbf{X})$. Similarly, we transform the external memory \mathbf{M} into key $K(\mathbf{M})$ and value $V(\mathbf{M})$, and both are implemented as MLPs. The channel dimension of the key is L , while that of the value is U . The relevance between \mathbf{X} and each scene prototype \mathbf{p}_s can be measured by dot product similarity and a softmax function as follows:

$$R(\mathbf{X}, \mathbf{M}) = \text{softmax}\left(\frac{Q(f_\phi(\mathbf{X})) \cdot K(\mathbf{M})^T}{\sqrt{L}}\right). \quad (4)$$

The output is an S -dimensional vector, where each component represents a relevance probability that a specific scene prototype is related to the query image. Subsequently, the retrieved scene prototypes are computed by weight-summing all values with the following equation:

$$\mathbf{z} = R(\mathbf{X}, \mathbf{M}) \cdot V(\mathbf{M}). \quad (5)$$

Since the memory retrieval is designed in a multi-head fashion, the final retrieved prototype is reformulated as follows:

$$\mathbf{z}' = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_H], \quad (6)$$

where H denotes the number of heads, and each head yields a retrieved prototype \mathbf{z}_h by transforming \mathbf{X} and \mathbf{M} to the variant query $Q_h(f_\phi(\mathbf{X}))$, key $K_h(\mathbf{M})$, and value $V_h(\mathbf{M})$. Eventually, the output \mathbf{z}' is fed into a fully-connected layer followed by a sigmoid function for inferring presences of aerial scenes.

3.4. Implementation Details

For a comprehensive assessment of our PM-Net, we implement the embedding function with various backbone CNNs. Specifically, we conduct experiments on four CNN architectures, and details are as follows:

- PM-VGGNet: f_ϕ is built on VGG-16 [65] by replacing all layers after the last max-pooling layer in *block5* with a global average pooling layer.
- PM-Inception-V3: Inception-V3 [66] is utilized, and layers before and including the global average pooling layer are employed as f_ϕ .

- PM-ResNet: We modify ResNet-50 [59] by discarding layers after the global average pooling layer and using the remaining layers as f_ϕ .
- PM-NASNet: The backbone of f_ϕ is mobile NASNet [67]. As with the modification in PM-ResNet, only layers before and including the global average pooling layer are used.

In our experiments, we train original deep CNNs on single-scene aerial image datasets and then take them as the embedding function f_ϕ following the aforementioned points. Subsequently, we yield scene prototypes \mathbf{p}_s and concatenate all of them along the first axis to form \mathbf{M} .

4. Experiments and Discussion

In this section, we introduce a newly produced multi-scene aerial image dataset, MAI dataset, and two single-scene datasets, i.e., UCM and AID datasets, which are used in experiments. Then network configurations and training schemes are detailed in Subsection 4.2. The remaining subsections discuss and analyze the performance of the proposed network thoroughly.

4.1. Dataset Description and Configuration

4.1.1. MAI dataset

To facilitate the progress of aerial scene interpretation in the wild, we yield a new dataset, MAI dataset, by collecting and labeling 3923 large-scale images from Google Earth imagery that covers the United States, Germany, and France. The size of each image is 512×512 , and spatial resolutions vary from 0.3 m/pixel to 0.6 m/pixel. After capturing aerial images, we manually assign each image multiple scene-level labels from in total 24 scene categories, including apron, baseball, beach, commercial, farmland, woodland, parking lot, port, residential, river, storage tanks, sea, bridge, lake, park, roundabout, soccer field, stadium, train station, works, golf course, runway, sparse shrub, and tennis court. Notably, OSM data associated with the collected images cannot be directly employed as reference owing to the problems presented in Section 1. Such a labeling procedure is extremely time- and labor-consuming, and annotating one image costs around 20 seconds, which is ten times more than labeling a single-scene image. Several example multi-scene images are shown in Figure 4. Numbers of aerial images related to various scenes are reported in Figure 5. Among existing datasets, BigEarthNet [68] is one of



Figure 4: Example images in our MAI dataset. Each image is 512×512 pixels, and their spatial resolutions range from 0.3 m/pixel to 0.6 m/pixel. We list their scene-level labels here: (a) farmland and residential; (b) baseball, woodland, parking lot, and tennis court; (c) commercial, parking lot, and residential; (d) woodland, residential, river, and runway; (e) river and storage tanks; (f) beach, woodland, residential, and sea; (g) farmland, woodland, and residential; (h) apron and runway; (i) baseball field, parking lot, residential, bridge, and soccer field.

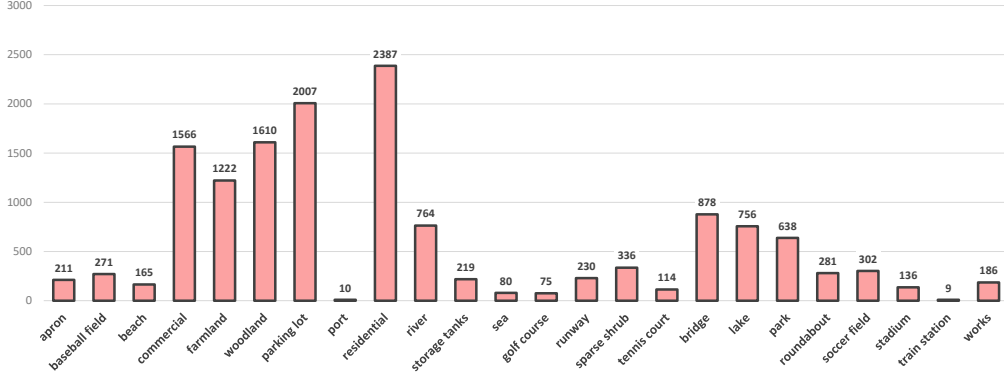


Figure 5: Statistics of the proposed MAI dataset for multi-scene classification in single aerial images.

the most relevant datasets, which consists of Sentinel-2 images acquired over the European Union with spatial resolutions ranging from 10 m/pixel to 60 m/pixel. Spatial sizes of images vary from 20×20 pixels to 120×120 pixels, and each is assigned multiple land-cover labels provided from the CORINE Land Cover map². Compared to BigEarthNet, our dataset is characterized by its high-resolution large-scale aerial images and worldwide coverage.

4.1.2. UCM dataset

UCM dataset [69] is a commonly used single-scene aerial image dataset produced by Yang and Newsam from the University of California Merced. This dataset comprises 2100 aerial images cropped from aerial ortho imagery provided by the United States Geological Survey (USGS) National Map, and the spatial resolution of the collected images is one foot. The size of each image is 256×256 pixels, and all image samples are classified into 21 scene-level classes: overpass, forest, beach, baseball diamond, building, airplane, freeway, intersection, harbor, golf course, runway, agricultural, storage tank, mobile home park, medium residential, sparse residential, chaparral, river, tennis courts, dense residential, and parking lot. The number of aerial images collected for each scene is 100, and several example images are shown in Figure 6. To learn scene prototypes from these single-scene images, we randomly choose 80% of image samples per scene category to train and validate

²<https://land.copernicus.eu/pan-european/corine-land-cover>



Figure 6: Example single-scene aerial categories in the UCM dataset: (a) agricultural, (b) dense residential, (c) forest, (d) storage tanks, (e) baseball field, (f) parking lot, (g) river, (h) runway, (i) golf course, and (j) tennis court.

the embedding function and utilize the rest for testing.

4.1.3. AID dataset

AID dataset [30] is a another popular single-scene aerial image dataset which consists of 10000 aerial images with a size of 600×600 pixels. These images are captured from Google Earth imagery that is taken over China, the United States, England, France, Italy, Japan, and Germany, and spatial resolutions of the collected images vary from 0.5 m/pixel to 8 m/pixel. In total, there are 30 scene categories, including viaduct, river, baseball field, center, farmland, railway station, meadow, bare land, storage tanks, beach, mountain, park, bridge, playground, church, commercial, desert, forest, parking, industrial, square, sparse residential, pond, medium residential, port, resort, airport, school, stadium, and dense residential. The number of images in different classes ranges from 220 to 420. Similar to the data split in the UCM dataset, 20% of images are chosen from each scene as test samples, while the remaining images are utilized to train and validate the embedding function. Some example images of the AID dataset are exhibited in Figure 7.

4.1.4. Dataset configuration

In order to widely evaluate the performance of our method, we utilize two variant dataset configurations, UCM2MAI and AID2MAI, based on common scene categories shared by UCM/AID and MAI. Specifically, the UCM2MAI configuration consists of 1600 single-scene aerial images from the UCM dataset and 1649 multi-scene images from our MAI dataset. 16 aerial scenes that are commonly included in both two datasets are considered in UCM2MAI, and numbers of their associated images are listed in Table 1. Besides, the AID2MAI configuration is composed of 7050 and 3239 aerial images from the AID and MAI datasets, respectively. 20 common scene categories are taken into consideration, and the number of images related to each scene is present in Table 1. Although such configurations might limit the number of recognizable scene classes, we believe this limitation can be addressed by collecting more single-scene images by crawling OSM data and producing large-scale multi-scene aerial image datasets. We select only 90 and 120 multi-scene aerial images from UCM2MAI and AID2MAI as training instances, respectively, and test networks on the remaining multi-scene images. For rare scenes (e.g., port and train station), we select all associated training images, while for common scenes, we randomly select several of their training samples. It is noteworthy that we yield the scene prototype of **residential** by taking an average of high-level representations of aerial images belonging to scene **medium residential** and **dense residential**. Besides, although the UCM and AID datasets do not contain images for **sea**, their images for **beach** often comprise both sea and beach (cf. (c) in Figure 7). Therefore, we make use of training samples labeled as **beach** to yield the prototype representation of **sea**.

4.2. Training Details

The training procedure consists of two phases: 1) learning the embedding function f_ϕ on large quantities of single-scene aerial images and 2) training the entire PM-Net on a limited number of multi-scene images in an end-to-end manner. Thus, various training strategies are applied to each phase and detailed as follows.

In the first training phase, the embedding function f_ϕ is initialized with the corresponding deep CNNs pretrained on ImageNet [70], and weights in g_θ are initialized by a Glorot uniform initializer. Eq. (1) is employed as the loss of the network, and Nestrov Adam [71] is chosen as the optimizer, of which parameters are set as recommended: $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e - 08$.

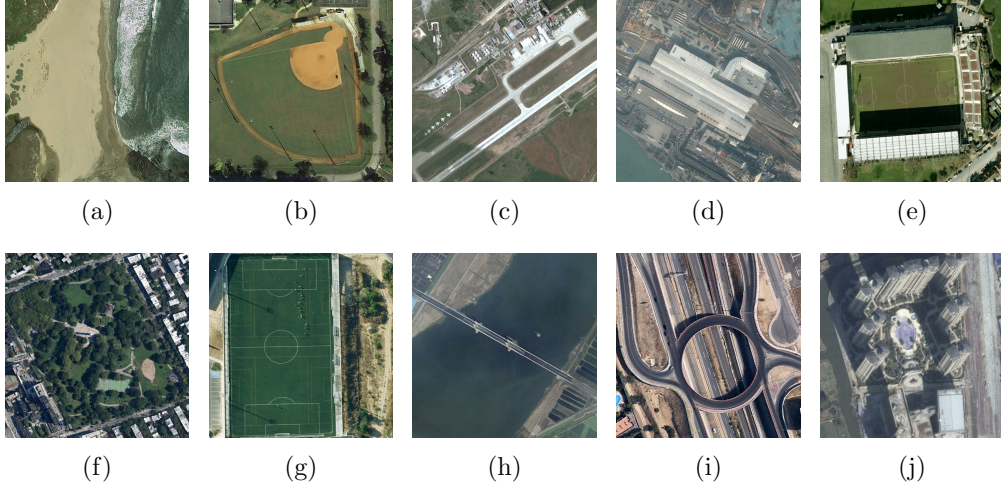


Figure 7: Example single-scene aerial categories in the AID dataset: (a) beach, (b) baseball field, (c) airport, (d) railway station, (e) stadium, (f) park, (g) playground, (h) bridge, (i) viaduct, and (j) commercial.

The learning rate is set as $2e - 04$ and decayed by $\sqrt{0.1}$ when the validation loss fails to decrease for two epochs.

In the second learning phase, we initialize f_ϕ with parameters learned in the previous training stage and employ the Glorot uniform initializer to initialize all weights in Q_h , V_h , K_h , and the last fully-connected layer. L and U are set to the same value of 256, and the number of heads is defined as 20. Notably, all weights are trainable, and the embedding function is tuned during the second training phase as well. Multiple scene-level labels are encoded as multi-hot vectors, where 0 indicates the absence of the corresponding scene while 1 refers to existing scenes. Accordingly, the loss is defined as binary cross-entropy. The optimizer is the same as that in the first training phase, but here we make use of a relatively large learning rate, $5e - 4$. The network is implemented on TensorFlow and trained on one NVIDIA Tesla P100 16GB GPU for 100 epochs. We set the size of training batch to 32 for both training phases.

4.3. Evaluation Metrics

For the purpose of evaluating the performance of networks quantitatively, we utilize example-based F_1 [72] and F_2 [73] scores as evaluation metrics and

Table 1: The Number of Images Associated with Each Scene.

	UCM2MAI		AID2MAI	
Scene Category	UCM	MAI	AID	MAI
apron	100	194	360	54
baseball field	100	75	220	235
beach	100	94	400	130
commercial	100	607	350	1391
farmland	100	680	370	983
woodland	100	762	250	1312
parking lot	100	708	390	1777
port	100	3	380	9
residential	200	958	700	2082
river	100	209	410	686
storage tanks	100	89	360	193
sea	100*	51	400*	59
golf course	100	75	-	-
runway	100	230	-	-
sparse shrub	100	336	-	-
tennis court	100	114	-	-
bridge	-	-	360	878
lake	-	-	420	756
park	-	-	350	638
roundabout	-	-	420	281
soccer field	-	-	370	302
stadium	-	-	290	136
train station	-	-	260	9
works	-	-	390	186
All	1600	1649	7050	3239

* indicates that the number of images is not counted in total amounts, as the scene prototype of **beach** and **sea** are learned from the same images.

calculate them with the following equation:

$$F_\beta = (1 + \beta^2) \frac{p_e r_e}{\beta^2 p_e + r_e}, \quad \beta = 1, 2, \quad (7)$$

Table 2: Differences between Two Training Phases.

Phase	Learnable Module	Dataset		Memory
		Pretraining f_ϕ	Fine-tuning module	
1	prototype learning	ImageNet	UCM/AID	updated
2	memory retrieval	UCM/AID	MAI	frozen

where p_e and r_e denote example-based precision and recall [74]. We calculate p_e and r_e as follows:

$$p_e = \frac{TP_e}{TP_e + FP_e}, \quad r_e = \frac{TP_e}{TP_e + FN_e}, \quad (8)$$

where FN_e , FP_e , and TP_e represent numbers of false negatives, false positives, and true positives in an example, respectively. In our case, an example is a multi-scene aerial image, and by averaging scores of all examples in the test set, the mean example-based F scores, precision, and recall can be eventually computed. In addition to example-based evaluation metrics, we also calculate label-based precision p_l and recall r_l with Eq. 8 but replace FN_e , FP_e , and TP_e with numbers of false negatives, false positives, and true positives in respect of each scene category. The mean p_l and r_l can then be calculated. Note that principle indexes are the mean F_1 and F_2 scores.

4.4. Results on UCM2MAI

For a comprehensive evaluation, we compare the proposed PM-Net with two baselines, CNN* and CNN. The former is initialized with parameters pre-trained on ImageNet, and the latter is pretrained on single-scene datasets. Besides, we compare our network with a memory network, Mem-N2N [47]. Since Mem-N2N was proposed for the question answering task, we adapt it to our task by replacing its inputs, i.e., embeddings of *questions* and *statements*, with *query image representations* $f_\phi(\mathbf{X})$ and *scene prototypes* \mathbf{p}_s , respectively. To be more specific, we feed \mathbf{X} to a CNN backbone and take its output as the input of Mem-N2N. Scene prototypes are stored in the memory of Mem-N2N and retrieved according to $f_\phi(\mathbf{X})$. The initialization of f_ϕ is the same as that of our network, and the entire Mem-N2N is trained in an end-to-end manner. Various backbones of embedding functions are test, and quantitative results are reported in Table 3. Besides, we also compare

Table 3: Numerical Results on UCM2MAI (%).

Model	m. F_1	m. F_2	m. p_e	m. r_e	m. p_l	m. r_l
VGGNet* [65]	32.16	32.79	35.08	34.35	21.74	22.57
VGGNet [65]	51.42	49.04	62.00	48.38	36.80	27.44
Mem-N2N-VGGNet [47]	52.16	50.93	57.26	50.73	20.79	22.58
K-Branch CNN [36]	47.04	43.15	64.57	41.83	37.93	22.28
proposed PM-VGGNet	54.42	51.16	67.35	49.95	47.24	26.79
Inception-V3* [66]	48.03	44.37	62.22	42.80	47.36	20.43
Inception-V3 [66]	53.96	51.28	65.47	50.49	51.03	32.88
Mem-N2N-Inception-V3 [47]	56.06	55.27	62.95	55.92	47.90	30.48
proposed PM-Inception-V3	58.56	58.06	64.17	58.73	46.44	26.47
ResNet* [59]	48.36	45.00	63.90	43.84	53.63	28.35
ResNet [59]	51.39	48.31	65.33	47.37	51.89	30.54
Mem-N2N-ResNet [47]	54.31	51.45	63.97	50.31	44.33	24.58
proposed PM-ResNet	56.89	54.11	69.85	53.38	55.93	29.76
NASNet* [67]	43.64	39.94	58.56	38.39	46.01	19.69
NASNet [67]	52.03	49.43	64.24	48.75	49.99	33.75
Mem-N2N-NASNet [47]	55.17	53.05	64.71	52.65	49.60	29.14
proposed PM-NASNet	60.13	59.57	67.04	60.42	58.60	35.04

CNN* is initialized with weights pretrained on ImageNet.

CNN, Mem-N2N, and PM-Net are initialized with parameters pretrained on the UCM dataset.

m. F_1 and m. F_2 indicate the mean F_1 and F_2 score.

m. p_e and m. r_e indicate mean example-based precision and recall.

m. p_l and m. r_l indicate mean label-based precision and recall.

with a multi-attention driven multi-label classification network, termed as K-Branch CNN [36]. K-Branch samples images into K spatial resolutions and extracts their features with separate branches. Afterwards, a bidirectional recurrent neural network is employed to encode their relationships for

Table 4: Example Images and Predictions on UCM2MAI.

Sample Multi-scene Aerial Images from MAI Dataset				
Ground Truths	farmland, woodland, residential	commercial, parking lot, residential	woodland, farmland	commercial, beach, parking lot, residential
Predictions	farmland, woodland, residential	commercial, parking lot, residential	woodland, farmland	commercial, beach, parking lot, residential

Sample Multi-scene Aerial Images from MAI Dataset				
Ground Truths	farmland, parking lot, residential	baseball field, parking lot, residential, tennis court	beach, parking lot, woodland, residential, sea	apron, runway
Predictions	farmland, parking lot, residential	baseball field, parking lot, residential, tennis court	commercial, beach, parking lot, woodland, residential, sea	apron, residential, runway, parking lot

Blue predictions are false negatives, while red predictions indicate false positives.

inferring multiple labels. In our experiments, K is set as default, 3, and input sizes of the three branches are 224×224 , 112×112 , and 56×56 , respectively. Here we analyze results from the following three perspectives.

Table 5: Numerical results on AID2MAI (%).

Model	m. F_1	m. F_2	m. p_e	m. r_e	m. p_l	m. r_l
VGGNet* [65]	41.57	36.36	64.02	34.04	25.98	12.80
VGGNet [65]	48.30	50.80	48.53	54.19	32.89	44.75
Mem-N2N-VGGNet [47]	45.92	43.17	56.16	42.22	23.10	18.76
K-Branch CNN [36]	47.67	43.88	63.84	42.37	26.53	16.15
proposed PM-VGGNet	54.37	51.44	65.69	50.39	48.06	22.40
Inception-V3* [66]	45.92	40.76	66.17	38.43	39.56	14.71
Inception-V3 [66]	51.81	49.44	62.91	48.93	45.26	36.32
Mem-N2N-Inception-V3 [47]	52.13	53.83	52.53	56.21	33.33	29.05
proposed PM-Inception-V3	53.08	49.26	69.42	47.85	48.20	24.65
ResNet* [59]	50.06	46.88	64.32	45.98	39.48	22.34
ResNet [59]	54.74	52.76	65.54	52.62	47.54	40.23
Mem-N2N-ResNet [47]	53.26	60.41	46.15	68.07	23.75	30.21
proposed PM-ResNet	57.42	54.34	70.62	53.33	55.34	29.55
NASNet* [67]	47.53	42.93	65.57	40.94	34.79	16.42
NASNet [67]	53.08	50.68	64.33	50.17	46.68	37.43
Mem-N2N-NASNet [47]	39.27	40.72	38.52	42.38	20.03	20.41
proposed PM-NASNet	54.11	52.39	64.03	52.30	43.16	33.99

CNN, Mem-N2N, and PM-Net are initialized with parameters pretrained on the AID dataset.

4.4.1. The effectiveness of learnt single-scene prototypes

To demonstrate the effectiveness of the prototype-inhabiting external memory, here we focus on comparisons between PM-Net and standard CNNs. In Table 3, PM-VGGNet increases the mean F_1 and F_2 scores by 3.00% and 2.12%, respectively, with respect to VGGNet, and PM-ResNet obtains increments of 5.50% and 5.80% in the mean F_1 and F_2 scores compared to ResNet. Besides, it is interesting to observe that PM-NASNet achieves not only the best mean F_1 and F_2 scores (60.13% and 59.57%) but also relatively high example-based precision and recall in comparison with other competitors.

Table 6: Example Images and Predictions on AID2MAI.

Sample multi-scene aerial images from MAI dataset				
Ground Truths	bridge, river, commercial, parking lot, residential	beach, commercial, parking lot, residential, sea	bridge, farmland, river, woodland	baseball field, parking lot, park, residential
Predictions	bridge, river, commercial, parking lot, residential	beach, commercial, parking lot, residential, sea	bridge, farmland, river, woodland	baseball field, parking lot, park, residential
Sample multi-scene aerial images from MAI dataset				
Ground Truths	beach, commercial, parking lot, residential, sea	bridge, woodland, river, storage tanks	baseball field, commercial, parking lot, park, residential, soccer field	baseball field, parking lot, soccer field
Predictions	beach, commercial, parking lot, residential, sea, woodland	bridge, woodland, farmland, river, storage tanks, parking lot	baseball field, commercial, woodland, parking lot, park, soccer field, residential	baseball field, commercial, woodland, parking lot, park, soccer field, residential

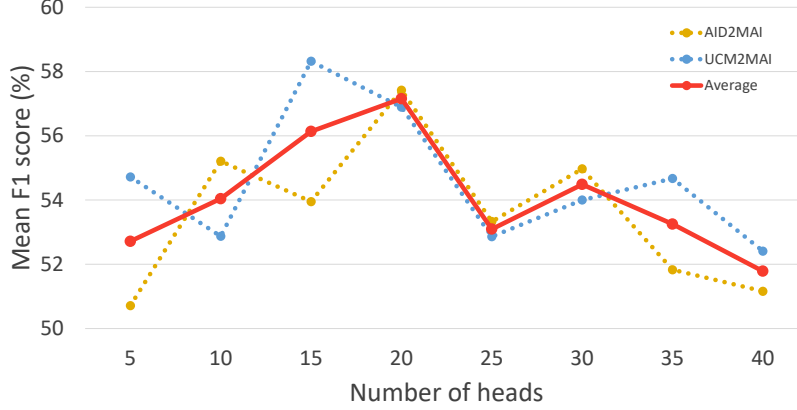


Figure 8: The influence of the number of heads on both dataset configurations. Blue and yellow dot lines represent mean F_1 scores on UCM2MAI and AID2MAI. The Red line indicates the average of them.

This demonstrates that employing NASNet as the embedding function can enhance the robustness of PM-Net. Comparisons between PM-Inception-V3 with Inception-V3 show that the external memory module contributes to improvements of 4.60% and 6.78% in the mean F_1 and F_2 scores, respectively. To summarize, memorizing and leveraging scene prototypes learned from huge quantities of single-scene images can improve the performance of network in multi-label scene recognition when limited training samples are available. For a deep insight, we further conduct ablation studies on the prototype modality and embedding function.

Single- vs. multi-prototype representations. We note that images collected over variant countries show high intra-class variability, and therefore, we wonder whether learning multi-prototype scene representations could improve the effectiveness of PM-Net. Specifically, instead of yielding scene prototypes via Eq. 2, we partition representations of single-scene aerial images belonging to the same scene into several clusters and take cluster centers as multi-prototype representations of each scene. In our experiments, we test two clustering methods, K-Means [75] and Agglomerative [76], with PM-ResNet on both UCM2MAI and AID2MAI, and results are shown in Figure 9. We can see that the performance of PM-ResNet is decreased with the increasing number of cluster centers either using K-Means or Agglomerative clustering algorithms. Explanations could be that there are no obvious subclusters within each scene category (cf. Figure 13), and thus PM-Net

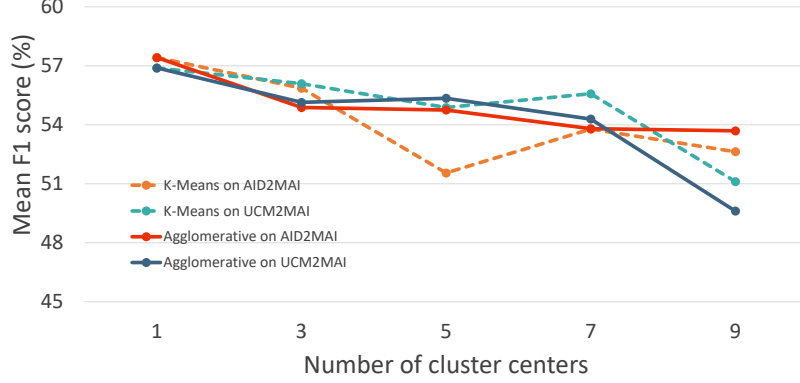


Figure 9: The influence of the number of cluster centers on both dataset configurations. K-Means (turquoise and orange dash lines) and Agglomerative (blue and red lines) clustering algorithms are tested with PM-ResNet on both UCM2MAI and AID2MAI, respectively.

does not benefit from fine-grained multi-prototype representations.

Frozen vs. trainable embedding function. The embedding function plays a key role in both scene prototype learning and memory retrieval. In the former, we train the embedding function on single-scene images, while in the latter, the function is fine-tuned on multi-scene images. To explore the effectiveness of fine-tuning, we conduct experiments on freezing the embedding function when learning the memory retrieval module. The comparisons between PM-Net learned with frozen and trainable embedding functions are shown in Figure 10. It can be observed that PM-Net with a trainable embedding function shows higher performance on both UCM2MAI and AID2MAI configurations. The reason could be that sources of single- and multi-scene images are variant, and fine-tuning can narrow their gaps.

Triplet vs. cross-entropy loss. Triplet loss [77] is known as learning discriminative representations by minimizing distances between embeddings of the same class while pushing away those of different classes. To study its performance in our task, we train the embedding function by replacing Eq. 1 with the following equation:

$$\mathcal{L}(\mathbf{X}_i^s) = \max(\|f_\phi(\mathbf{X}_i^s) - f_\phi(\mathbf{X}_{pos}^s)\|^2 - \|f_\phi(\mathbf{X}_i^s) - f_\phi(\mathbf{X}_{neg}^s)\|^2 + \alpha, 0), \quad (9)$$

where \mathbf{X}_{pos}^s and \mathbf{X}_{neg}^s denote positive and negative samples, i.e., images belonging to common and different classes, respectively, and α is set as default, 0.5. The trained embedding function is then utilized to extract scene prototypes and initialize f_ϕ in the phase of learning the memory retrieval module.

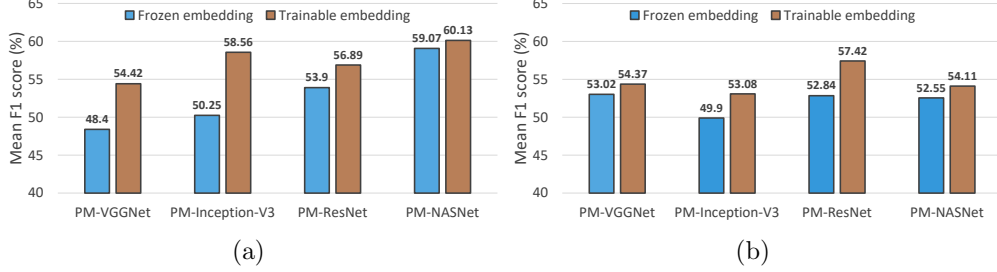


Figure 10: Comparisons between freezing and fine-tuning embedding functions on (a) UCM2MAI and (b) AID2MAI, respectively. Blue bars represent the performance of PM-Net with frozen embedding functions, and brown bars denote the performance of PM-Net with trainable embedding functions.

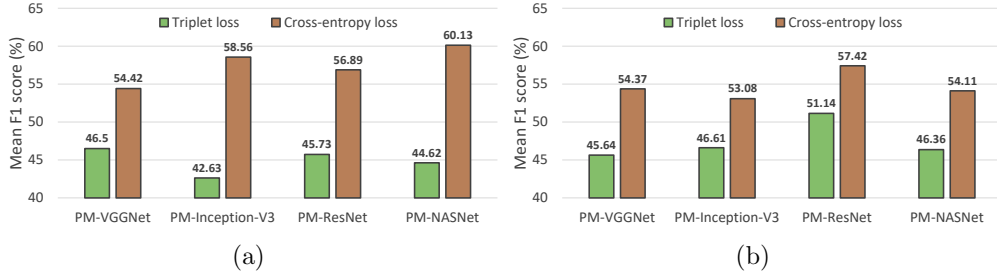


Figure 11: Comparisons of different loss functions on (a) UCM2MAI and (b) AID2MAI, respectively. Green bars denote the performance of PM-Net using embedding functions trained by the triplet loss, and brown bars denote the performance of PM-Net with the cross-entropy loss as \mathcal{L} .

Besides, all the other setups are remained the same. We compare the performance of PM-Net using embedding functions trained through different loss functions in Figure 11. It can be seen that training embedding functions with the triplet loss leads to decrements of the network performance. This can be attributed to that limited numbers of positive and negative samples in each batch can lead to local optimum. More specifically, the size of training batches is 32, and the number of scenes are 16 and 20 in UCM2MAI and AID2MAI, respectively. Thus, it is high probably that only a certain number of scenes are included in one batch, and comprehensively modeling relations between embeddings of samples from all scenes is infeasible. This also illustrates the larger performance decay on UCM2MAI compared to AID2MAI.

4.4.2. The effectiveness of our multi-head attention-based memory retrieval module

As a key component of the proposed PM-Net, the multi-head attention-based memory retrieval module is designed to retrieve scene prototypes from the external memory, and we evaluate its effectiveness by comparing PM-Net with Mem-N2N. As shown in Table 3, PM-Net outperforms Mem-N2N with variant embedding functions. Specifically, PM-VGGNet increases the mean F_1 and F_2 scores by 2.26% and 0.23%, respectively, compared to Mem-N2N-VGGNet. While taking ResNet as the embedding function, the improvement can reach 2.58% in the mean F_1 score. Besides, the highest increments of mean F_1 and F_2 scores, 4.96% and 6.52, are achieved by PM-NASNet. These observations demonstrate that our memory retrieval module plays a key role in inferring multiple aerial scenes. An explanation could be that compared to the memory reader in Mem-N2N, our module comprise multiple heads, and each of them focuses on encoding a specific relevance between the query image and variant scene prototypes. In this case, more comprehensive scene-related memories can be used for inferring multiple scene labels. Moreover, we analyze the influence of the number of heads in the memory retrieval module. Figure 8 shows mean F_1 scores achieved by PM-Net with variant head numbers on both UCM2MAI and AID2MAI. We can observe that the network performance is first boosted with an increasing number of heads and then decreased gradually when the number exceeds 20.

Moreover, we also conduct experiments on directly utilizing relevances for inferring multiple scene labels. Specifically, we set the number of heads to 1 and replace the softmax activation in Eq. 4 with the sigmoid function. Relevances between the query image and scene prototypes can then be interpreted as the existence of each scene. We compare it with our memory retrieval module on variant backbones, and results are shown in Figure 12. We can see that utilizing relevances $R(\mathbf{X}, \mathbf{M})$ as weights for aggregating scene prototypes leads to higher network performance.

4.4.3. The benefit of exploiting single-scene training samples

Let’s start with the conclusion: exploiting single-scene images significantly contributes to our task. To analyze its benefit, we mainly compare CNNs* and CNNs. It can be observed that even with identical network architectures, the performance of CNN is superior to that of CNN*. More specifically, VGGNet achieves the highest improvement of the mean F_1 scores, 19.26%, in comparison with VGGNet*. NASNet shows higher performance

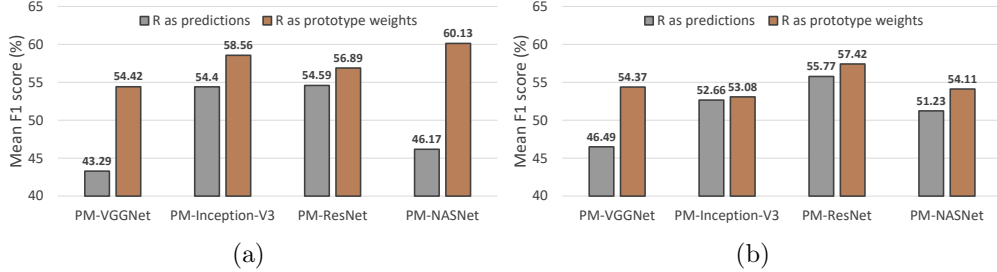


Figure 12: Comparisons between taking relevance $R(\mathbf{X}, \mathbf{M})$ as predictions and prototype weights on (a) UCM2MAI and (b) AID2MAI, respectively. Gray and brown bars represent the performance of PM-Net making predictions from relevances and aggregated scene prototypes, respectively.

in all metrics compared to ResNet*, while other CNNs perform poorly in only the mean example-based precision with respect to their corresponding CNNs*. Besides, we visualize features of single-scene images learned by VGGNet on UCM and AID datasets via t-SNE, respectively. As shown in Figure 13, extracted features are discriminative and separable in the embedding space, which demonstrates the effectiveness of learning the embedding function on single-scene aerial image datasets. To summarize, except for learning scene prototypes, single-scene training samples can also benefit multi-label scene interpretation by pretraining CNNs which are further utilized to initialize the embedding function.

We exhibit several example predictions of PM-ResNet trained on UCM2MAI in Table 4. False positives are marked as red, while false negatives are in blue. As shown in the forth example at the top row, we see that PM-Net can accurately perceive aerial scenes even in complex contexts, but unseen scene appearance (i.e. apron and runway in snow) can influence its prediction.

4.5. Results on AID2MAI

Table 5 reports numerical results on the AID2MAI configuration. It can be seen that the performance of PM-Net is superior to all competitors in the mean F_1 score. Compared to Mem-N2N-VGGNet, the proposed PM-VGGNet increases the mean F_1 and F_2 scores by 6.70% and 7.56%, respectively, while improvements reach 6.07% and 0.64% in comparison with VGGNet. PM-ResNet achieves the best mean F_1 score and example-based precision, 57.42% and 70.62, respectively. With NASNet as the backbone, exploiting the proposed memory retrieval module contributes to increments

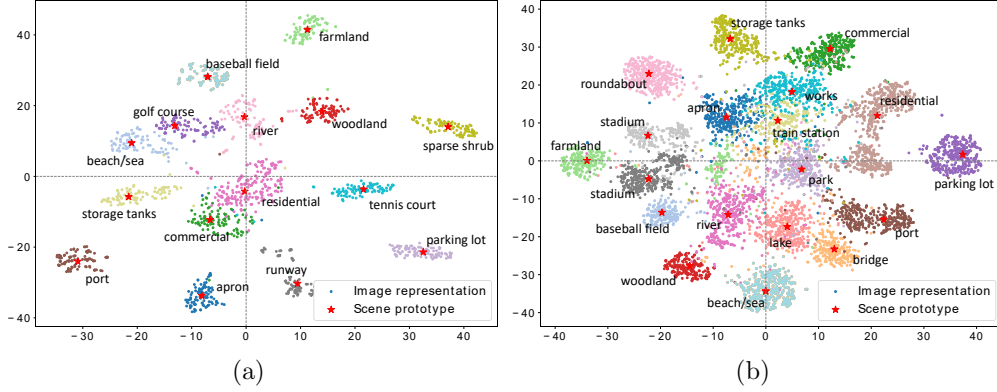


Figure 13: T-SNE visualization of image representations and scene prototypes learned by VGGNet on (a) UCM and (b) AID datasets, respectively. Dots in the same color represent features of images belonging to the same scene, and stars denote scene prototypes.

of 1.03% and 1.71% in mean F_1 and F_2 scores compared to directly learning NASNet on a small number of multi-scene samples.

We present some example predictions of PM-ResNet in Table 6. As shown in the top row, PM-ResNet learned with a limited number of annotated multi-scene images can accurately identify various aerial scenes even image contextual information is complicated. The bottom row shows some inaccurate predictions. It can be observed that although bridge and parking lot account for relatively small areas in last two examples at the top row, the proposed PM-Net can successfully detect them. Similar observations can also be found in the first and third example at the bottom row that residential and parking lot are recognized by our network, even they are located at the corner. In conclusion, quantitative results illustrate the effectiveness of our network in learning to perform unconstrained multi-scene classification, and example predictions further demonstrate it.

5. Conclusion

In this paper, we propose a novel multi-scene recognition network, namely PM-Net, to tackle both the problem of aerial scene classification in the wild and scarce training samples. To be more specific, our network consists of three key elements: 1) a prototype learning module for encoding prototype representations of variant aerial scenes, 2) a prototype-inhabiting

external memory for storing high-level scene prototypes, and 3) a multi-head attention-based memory retrieval module for retrieving associated scene prototypes from the external memory for recognizing multiple scenes in a query aerial image. For the purpose of facilitating the progress as well as evaluating our method, we propose a new dataset, MAI dataset, and experiment with two dataset configurations, UCM2MAI and AID2MAI, based on two single-scene aerial image datasets, UCM and AID. In scene prototype learning, we train the embedding function on most of single-scene images as we aim to simulate the real-life scenario, where massive single-scene samples can be collected at low cost by resorting to OSM data. To learn memory retrieval, our network is fine-tuned on only around 100 training samples from the MAI dataset. Experimental results on both UCM2MAI and AID2MAI illustrate that learning and memorizing scene prototypes with our PM-Net can significantly improve the classification accuracy. The best performance is achieved by employing ResNet as the embedding function, and the best mean F_1 score reaches nearly 0.6. We hope that our work can open a new door for further researches in a more complicated and challenging task, multi-scene interpretation in single images. Looking into the future, we intend to apply the proposed network to the recovery of weakly supervised scenes.

Acknowledgements

This work is jointly supported by the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No. [ERC-2016-StG-714087], Acronym: *So2Sat*), by the Helmholtz Association through the Framework of Helmholtz AI [grant number: ZT-I-PF-5-01] - Local Unit “Munich Unit @Aeronautics, Space and Transport (MASTr)” and Helmholtz Excellent Professorship “Data Science in Earth Observation - Big Data Fusion for Urban Research” and by the German Federal Ministry of Education and Research (BMBF) in the framework of the international future AI lab “AI4EO – Artificial Intelligence for Earth Observation: Reasoning, Uncertainties, Ethics and Beyond” (Grant number: 01DD20001)

References

- [1] D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Datcu, U. Stilla, Classification with an edge: Improving semantic image seg-

- mentation with boundary detection, *ISPRS Journal of Photogrammetry and Remote Sensing* 135 (2018) 158–172.
- [2] N. Audebert, B. L. Saux, S. Lefèvre, Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks, *ISPRS Journal of Photogrammetry and Remote Sensing* 140 (2018) 20–32.
 - [3] D. Marcos, M. Volpi, B. Kellenberger, D. Tuia, Land cover mapping at very high resolution with rotation equivariant CNNs: Towards small yet accurate models, *ISPRS Journal of Photogrammetry and Remote Sensing* 145 (2018) 96–107.
 - [4] L. Mou, X. X. Zhu, RiFCN: Recurrent network in fully convolutional network for semantic segmentation of high resolution remote sensing images, arXiv:1805.02091.
 - [5] Q. Li, L. Mou, Q. Liu, Y. Wang, X. X. Zhu, HSF-Net: Multi-scale deep feature embedding for ship detection in optical remote sensing imagery, *IEEE Transactions on Geoscience and Remote Sensing*.
 - [6] C. Qiu, M. Schmitt, C. Geiß, T. Chen, X. X. Zhu, A framework for large-scale mapping of human settlement extent from Sentinel-2 images via fully convolutional neural networks, *ISPRS Journal of Photogrammetry and Remote Sensing* 163 (2020) 152–170.
 - [7] Q. Li, Y. Shi, X. Huang, X. X. Zhu, Building footprint generation by integrating convolution neural network with feature pairwise conditional random field (FPCRF), *IEEE Transactions on Geoscience and Remote Sensing*.
 - [8] L. Mou, X. X. Zhu, Vehicle instance segmentation from aerial image and video using a multi-task learning residual fully convolutional network, *IEEE Transactions on Geoscience and Remote Sensing* 56 (11) (2018) 6699–6711.
 - [9] L. Mou and X. X. Zhu, Spatiotemporal scene interpretation of space videos via deep neural network and tracklet analysis, in: *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2016.
 - [10] A. Vetrivel, M. Gerke, N. Kerle, F. Nex, G. Vosselman, Disaster damage detection through synergistic use of deep learning and 3D point

- cloud features derived from very high resolution oblique aerial images, and multiple-kernel-learning, *ISPRS Journal of Photogrammetry and Remote Sensing* 140 (2018) 45–59.
- [11] W. Lee, S. Kim, Y. Lee, H. Lee, M. Choi, Deep neural networks for wild fire detection with unmanned aerial vehicle, in: *IEEE International Conference on Consumer Electronics (ICCE)*, 2017.
 - [12] S. Lucchesi, M. Giardino, L. Perotti, Applications of high-resolution images and DTMs for detailed geomorphological analysis of mountain and plain areas of NW Italy, *European Journal of Remote Sensing* 46 (1) (2013) 216–233.
 - [13] Q. Weng, Z. Mao, J. Lin, X. Liao, Land-use scene classification based on a CNN using a constrained extreme learning machine, *International Journal of Remote Sensing* (2018) 1–19.
 - [14] G. Cheng, J. Han, X. Lu, Remote sensing image scene classification: Benchmark and state of the art, *Proceedings of the IEEE* 105 (10) (2017) 1865–1883.
 - [15] P. Zarco-Tejada, R. Diaz-Varela, V. Angileri, P. Loudjani, Tree height quantification using very high resolution imagery acquired from an unmanned aerial vehicle (UAV) and automatic 3D photo-reconstruction methods, *European Journal of Agronomy* 55 (2014) 89–99.
 - [16] D. Wen, X. Huang, H. Liu, W. Liao, L. Zhang, Semantic classification of urban trees using very high resolution satellite imagery, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 10 (4) (2017) 1413–1424.
 - [17] L. Mou, X. X. Zhu, IM2HEIGHT: Height estimation from single monocular imagery via fully residual convolutional-deconvolutional network, *arXiv:1802.10249*.
 - [18] C. Qiu, L. Mou, M. Schmitt, X. X. Zhu, Local climate zone-based urban land cover classification from multi-seasonal Sentinel-2 images with a recurrent residual network, *ISPRS Journal of Photogrammetry and Remote Sensing* 154 (2019) 151–162.

- [19] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, F. Fraundorfer, Deep learning in remote sensing: A comprehensive review and list of resources, *IEEE Geoscience and Remote Sensing Magazine* 5 (4) (2017) 8–36.
- [20] J. Murray, D. Marcos, D. Tuia, Zoom In, Zoom Out: Injecting scale invariance into landuse classification CNNs, in: *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019.
- [21] G. Cheng, X. Xie, J. Han, L. Guo, G. Xia, Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13 (2020) 3735–3756.
- [22] Q. Bi, K. Qin, Z. Li, H. Zhang, K. Xu, G. Xia, A multiple-instance densely-connected ConvNet for aerial scene classification, *IEEE Transactions on Image Processing* 29 (2020) 4911–4926.
- [23] S. Niazmardi, B. Demir, L. Bruzzone, A. Safari, S. Homayouni, Multiple kernel learning for remote sensing image classification, *IEEE Transactions on Geoscience and Remote Sensing* 56 (3) (2017) 1425–1443.
- [24] J. Lin, L. Mou, T. Yu, X. X. Zhu, Z. J. Wang, Dual adversarial network for unsupervised ground/satellite-to-aerial scene adaptation, in: *ACM International Conference on Multimedia (ACMMM)*, 2020.
- [25] Q. Zhu, Y. Zhong, L. Zhang, D. Li, Adaptive deep sparse semantic modeling framework for high spatial resolution image scene classification, *IEEE Transactions on Geoscience and Remote Sensing* 56 (10) (2018) 6180–6195.
- [26] A. Byju, G. Sumbul, B. Demir, L. Bruzzone, Remote sensing image scene classification with deep neural networks in JPEG 2000 compressed domain, *arXiv:2006.11529*.
- [27] Y. Xu, B. Du, L. Zhang, Assessing the threat of adversarial examples on deep neural networks for remote sensing scene classification: Attacks and defenses, *IEEE Transactions on Geoscience and Remote Sensing*.
- [28] X. Wang, X. Xiong, C. Ning, Multi-label remote sensing scene classification using multi-bag integration, *IEEE Access* 7 (2019) 120399–120410.

- [29] Q. Zhu, X. Sun, Y. Zhong, L. Zhang, High-resolution remote sensing image scene understanding: A review, in: IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2019.
- [30] G. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, X. Lu, AID: A benchmark data set for performance evaluation of aerial scene classification, IEEE Transactions on Geoscience and Remote Sensing.
- [31] P. Jin, G. Xia, F. Hu, Q. Lu, L. Zhang, AID++: An updated version of aid on scene classification, in: IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2018.
- [32] D. Hu, X. Li, L. Mou, P. Jin, D. Chen, L. Jing, X. X. Zhu, D. Dou, Cross-task transfer for multimodal aerial scene recognition, arXiv:2005.08449.
- [33] D. Tuia, D. Marcos, G. Camps-Valls, Multi-temporal and multi-source remote sensing image classification by nonlinear relative normalization, ISPRS Journal of Photogrammetry and Remote Sensing 120 (2016) 1–12.
- [34] L. Ru, B. Du, C. Wu, Multi-temporal scene classification and scene change detection with correlation based fusion, arXiv:2006.02176.
- [35] Q. Li, C. Qiu, L. Ma, M. Schmitt, X. X. Zhu, Mapping the land cover of Africa at 10 m resolution from multi-source remote sensing data with Google Earth engine, Remote Sensing 12 (4) (2020) 602.
- [36] G. Sumbul, B. Demir, A novel multi-attention driven system for multi-label remote sensing image classification, in: IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2019.
- [37] B. Zegeye, B. Demir, A novel active learning technique for multi-label remote sensing image scene classification, in: Image and Signal Processing for Remote Sensing, 2018.
- [38] Y. Hua, L. Mou, X. X. Zhu, Relation network for multilabel aerial image classification, IEEE Transactions on Geoscience and Remote Sensing.
- [39] N. Khan, U. Chaudhuri, B. Banerjee, S. Chaudhuri, Graph convolutional network for multi-label VHR remote sensing scene recognition, Neurocomputing 357 (2019) 36–46.

- [40] Y. Hua, L. Mou, X. X. Zhu, Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional LSTM network for multi-label aerial image classification, *ISPRS journal of photogrammetry and remote sensing* 149 (2019) 188–199.
- [41] A. Zeggada, F. Melgani, Y. Bazi, A deep learning approach to UAV image multilabeling, *IEEE Geoscience and Remote Sensing Letters* 14 (5) (2017) 694–698.
- [42] S. Koda, A. Zeggada, F. Melgani, R. Nishii, Spatial and structured SVM for multilabel image classification, *IEEE Transactions on Geoscience and Remote Sensing* 56 (10) (2018) 5948–5960.
- [43] Y. Long, G. Xia, S. Li, W. Yang, M. Yang, X. X. Zhu, L. Zhang, D. Li, DiRS: On creating benchmark datasets for remote sensing image interpretation, *arXiv:2006.12485*.
- [44] National Research Council, *How people learn: Brain, mind, experience, and school: Expanded edition*, 2000.
- [45] E. Liu, E. Mercado III, B. Church, I. Orduña, The easy-to-hard effect in human (*homo sapiens*) and rat (*rattus norvegicus*) auditory identification, *Journal of Comparative Psychology* 122 (2) (2008) 132.
- [46] I. McLaren, M. Suret, Transfer along a continuum: Differentiation or association, in: *Annual Conference of the Cognitive Science Society*, 2000.
- [47] S. Sukhbaatar, A. Szlam, J. Weston, R. Fergus, End-to-end memory networks, in: *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.
- [48] A. Miller, A. Fisch, J. Dodge, A. Karimi, A. Bordes, J. Weston, Key-value memory networks for directly reading documents, in: *Empirical Methods in Natural Language Processing (EMNLP)*, 2016.
- [49] Z. Shi, H. Fang, Y. Tai, C. Tang, DAWN: Dual augmented memory network for unsupervised video object tracking, *arXiv:1908.00777*.
- [50] H. Park, J. Noh, B. Ham, Learning memory-guided normality for anomaly detection, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

- [51] Z. Lai, E. Lu, W. Xie, MAST: A memory-augmented self-supervised tracker, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.
- [52] M. Cornia, M. Stefanini, L. Baraldi, R. Cucchiara, Meshed-memory transformer for image captioning, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.
- [53] J. Snell, K. Swersky, R. Zemel, Prototypical networks for few-shot learning, in: Advances in Neural Information Processing Systems (NeurIPS), 2017.
- [54] S. Guerriero, B. Caputo, M. T., DeepNCM: Deep nearest class mean classifiers, 2018.
- [55] H. Yang, X. Zhang, F. Yin, C. Liu, Robust classification with convolutional prototype learning, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
- [56] L. Huang, Y. Huang, W. Ouyang, L. Wang, Relational prototypical network for weakly supervised temporal action localization, in: AAAI Conference on Artificial Intelligence, 2020.
- [57] C. Zhang, J. Yue, Q. Qin, Global prototypical network for few-shot hyperspectral image classification, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 13 (2020) 4748–4759.
- [58] H. Tang, Y. Li, X. Han, Q. Huang, W. Xie, A spatial–spectral prototypical network for hyperspectral remote sensing image, IEEE Geoscience and Remote Sensing Letters 17 (1) (2019) 167–171.
- [59] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
- [60] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems (NeurIPS), 2017.
- [61] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, Improving language understanding by generative pre-training.

- [62] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, Language models are unsupervised multitask learners, 2019.
- [63] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, North American Chapter of the Association for Computational Linguistics.
- [64] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, A. Rush, Transformers: State-of-the-art natural language processing, in: Empirical Methods in Natural Language Processing: System Demonstrations, 2020.
- [65] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv:1409.1556.
- [66] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
- [67] B. Zoph, Q. Le, Neural architecture search with reinforcement learning, in: International Conference on Learning Representations (ICLR), 2017.
- [68] G. Sumbul, M. Charfuelan, B. Demir, V. Markl, BigEarthNet: A large-scale benchmark archive for remote sensing image understanding, in: IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2019.
- [69] Y. Yang, S. Newsam, Bag-of-visual-words and spatial extensions for land-use classification, in: International Conference on Advances in Geographic Information Systems (SIGSPATIAL), 2010.
- [70] J. Deng, W. Dong, R. Socher, L. Li, K. Li, F. Li, ImageNet: A large-scale hierarchical image database, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.
- [71] T. Dozat, Incorporating Nesterov momentum into Adam, http://cs229.stanford.edu/proj2015/054_report.pdf, online.

- [72] X. Wu, Z. Zhou, A unified view of multi-label performance measures, arXiv:1609.00288.
- [73] C. J. Van Rijsbergen, Information Retrieval, 1979.
- [74] G. Tsoumakas, I. Vlahavas, Random K-labelsets: An ensemble method for multilabel classification, in: European Conference on Machine Learning (ECML), 2007.
- [75] S. Lloyd, Least squares quantization in PCM, IEEE Transactions on Information Theory 28 (2) (1982) 129–137.
- [76] M. Zepeda-Mendoza, O. Resendis-Antonio, Hierarchical Agglomerative Clustering, 2013.
- [77] F. Schroff, D. Kalenichenko, J. Philbin, FaceNet: A unified embedding for face recognition and clustering, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

D Yuansheng Hua, Lichao Mou, Pu Jin and Xiao Xiang Zhu, “MultiScene: A large-scale dataset and benchmark for multiscene recognition in single aerial images,” *IEEE Transactions on Geoscience and Remote Sensing*, in press, 2021.

<https://doi.org/10.1109/TGRS.2021.3110314>

MultiScene: A Large-scale Dataset and Benchmark for Multi-scene Recognition in Single Aerial Images

Yuansheng Hua, Lichao Mou, Pu Jin, and Xiao Xiang Zhu, *Fellow, IEEE*

Abstract—This is the preprint version. To read the final version, please go to [IEEE Transactions on Geoscience and Remote Sensing](https://arxiv.org/abs/2012.04487). Aerial scene recognition is a fundamental research problem in interpreting high-resolution aerial imagery. Over the past few years, most studies focus on classifying an image into one scene category, while in real-world scenarios, it is more often that a single image contains multiple scenes. Therefore, in this paper, we investigate a more practical yet underexplored task—multi-scene recognition in single images. To this end, we create a large-scale dataset, called MultiScene, composed of 100,000 unconstrained high-resolution aerial images. Considering that manually labeling such images is extremely arduous, we resort to low-cost annotations from crowdsourcing platforms, e.g., OpenStreetMap (OSM). However, OSM data might suffer from incompleteness and incorrectness, which introduce noise into image labels. To address this issue, we visually inspect 14,000 images and correct their scene labels, yielding a subset of cleanly-annotated images, named MultiScene-Clean. With it, we can develop and evaluate deep networks for multi-scene recognition using clean data. Moreover, we provide crowdsourced annotations of all images for the purpose of studying network learning with noisy labels. We conduct experiments with extensive baseline models on both MultiScene-Clean and MultiScene to offer benchmarks for multi-scene recognition in single images and learning from noisy labels for this task, respectively. To facilitate progress, we make our dataset and trained models available on <https://gitlab.lrz.de/ai4eo/reasoning/multiscene>.

Index Terms—Convolutional neural network (CNN), multi-scene recognition in single images, crowdsourced annotations, large-scale aerial image dataset, learning from noisy labels

I. INTRODUCTION

With the recent development of Earth observation techniques, massive aerial imagery is now accessible for a variety of applications, such as environmental monitoring [1]–[6], urban planning [7]–[12], land cover and land use mapping [13]–[16], and disaster assessment [17], [18]. As one of the crucial

The work is jointly supported by the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No. [ERC-2016-StG-714087], Acronym: *So2Sat*), by the Helmholtz Association through the Framework of Helmholtz AI (grant number: ZT-I-PF-5-01) - Local Unit “Munich Unit @Aeronautics, Space and Transport (MASTr)” and Helmholtz Excellent Professorship “Data Science in Earth Observation - Big Data Fusion for Urban Research”(grant number: W2-W3-100) and by the German Federal Ministry of Education and Research (BMBF) in the framework of the international future AI lab “AI4EO – Artificial Intelligence for Earth Observation: Reasoning, Uncertainties, Ethics and Beyond” (grant number: 01DD20001). (Corresponding authors: Lichao Mou and Xiao Xiang Zhu.)

Y. Hua, L. Mou, and X. X. Zhu are with the Remote Sensing Technology Institute, German Aerospace Center, 82234 Weßling, Germany, and also with the Data Science in Earth Observation, Technical University of Munich, 80333 Munich, Germany. (e-mails: yuansheng.hua@dlr.de; lichao.mou@dlr.de; xiaoxiang.zhu@dlr.de)

P. Jin is with the Data Science in Earth Observation, Technical University of Munich, 80333 Munich, Germany. (e-mail: pu.jin@tum.de)

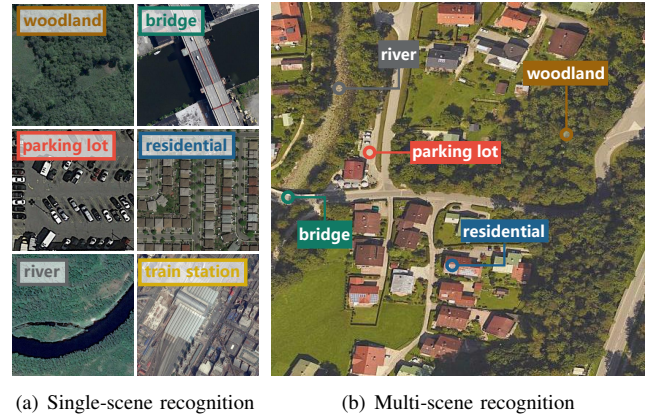


Fig. 1. Examples of images utilized in (a) single-scene and (b) multi-scene recognition tasks. In (a), each aerial image is assigned one scene label, while in (b), labels of all present scenes are inferred. In comparison with (b), (a) might suffer from partial scene understanding, as only one label is predicted even if there indeed exist multiple scenes in an image. For a clear visualization, locations of scenes are marked in (b).

steps towards these applications, aerial scene recognition has been extensively studied in the remote sensing community. During the last few years, the emergence of deep convolutional neural networks (CNNs) pushed ahead research in this field, and enormous achievements [19]–[26] have been obtained. Albeit successful, most existing scene classification researches only focus on a specific scenario, where an aerial image is assumed to include a single scene [27]–[34]. Basically, these studies regard aerial scene recognition as a single-label classification problem and learn models on well-cropped single-scene aerial images (see Fig. 1(a)). However, in practical applications, an aerial image often contains multiple scenes, as it is collected overhead and usually has a large coverage (cf. Fig. 1(b)). We also note that even in public single-scene aerial image datasets, the coexistence of multiple scenes in a single image is inevitable, especially in images covering large areas. For example, as shown in the bottom two images in Fig. 1(a), although they are assigned single scene labels according to their central/dominant scenes (i.e., river and train station), there actually exists more than one scene in each of them.

Hence, in this paper, we aim to tackle a more realistic yet challenging problem, namely multi-scene recognition in single aerial images. This task refers to assigning an aerial image multiple scene labels, and there are no constraints on image preparations, such as centering dominant scenes and eliminating clutter scenes. Compared to the conventional scene recognition task, multi-scene recognition is more arduous

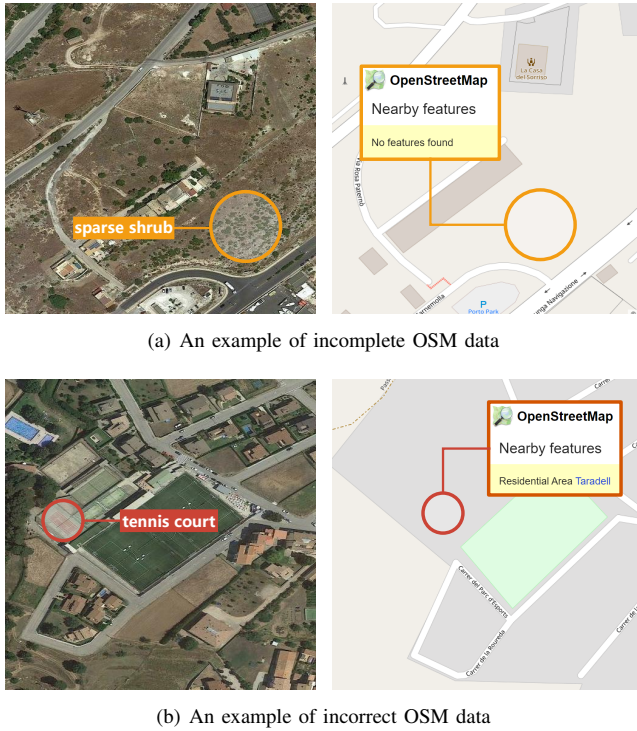


Fig. 2. Examples of (a) incomplete and (b) incorrect OSM annotations. In (a), sparse shrubs are not annotated in OSM data, while in (b), the tennis court is mislabeled as residential.

because 1) images are large-scale and unconstrained, and 2) all present scenes in an aerial image need to be exhaustively recognized. Fig. 1(b) shows an example of multi-scene aerial image and corresponding multiple scene-level labels. We can see that not only dominant scenes (e.g., residential and woodland) but also trivial scenes (e.g., bridge and parking lot) are annotated, which draws a more comprehensive picture for the unconstrained image.

However, very few efforts have been deployed to this problem in the remote sensing community. In order to advance the progress of multi-scene recognition in single images, we propose a large-scale Multi-Scene recognition (MultiScene) dataset, where 100,000 aerial images are collected around the world. In the phase of data preparation, we note that although massive high-resolution aerial images can be effortlessly obtained from remote sensing data platforms, such as Google Earth¹, it is extremely time- and labor-consuming to yield their corresponding multiple scene labels. To alleviate such annotation burden, in this paper, we resort to crowdsourced data, e.g., OpenStreetMap² (OSM) annotations, which has been proven to be successful in generating image-level labels [27], [28], [35] and pixel-wise footprints [12], [36] for training deep networks. However, we observe that OSM data might suffer from two common defects, incompleteness and incorrectness, which could introduce severe noise into image labels. Fig 2 shows two examples of incorrect OSM annotations, where (a) sparse shrubs are neglected, and (b) the tennis

court is mislabeled as residential. With this in mind, here we do not directly use crowdsourced labels as ground truth data. Instead, we visually inspect 14,000 images and correct their labels, producing a subset of cleanly-labeled images, named MultiScene-Clean. It allows developing and evaluating deep networks for unconstrained multi-scene recognition using clean data. Moreover, we note that the noisy crowdsourced data are not completely useless, for example, they can be used to study network learning with noisy labels for this task. Therefore, we also provide crowdsourced annotations of all images.

The contributions of this paper are four-fold:

- Unlike conventional aerial scene recognition where all images are well-cropped and each of them contains only one scene-level label, in this paper, we explore a more practical task—multi-scene recognition in single images.
- We propose a large-scale dataset, namely MultiScene, consisting of 100,000 unconstrained multi-scene aerial images, and each is assigned OSM labels. We visually inspect 14,000 images and correct their labels, yielding a subset of cleanly-labeled images.
- The proposed dataset provides not only ground truth data but also crowdsourced labels, which enables researches in learning from enormous noisy labels for our task.
- We extensively evaluate commonly-used classification networks on both MultiScene-Clean and MultiScene and provide benchmarks for recognizing multiple scenes in single images and learning from noisy labels for this task, respectively.

The remaining sections of this paper are organized as follows. Section II reviews studies in aerial single-scene classification and multi-label object classification. Section III briefly recalls existing scene datasets and delineates the proposed dataset. Experimental configurations and results are exhibited in Section IV, and Section V draws a conclusion.

II. RELATED WORK

This section briefly reviews related works in two fields: aerial single-scene classification and multi-label object recognition.

A. Aerial Single-scene Classification

Aerial single-scene classification refers to categorize an aerial image into a single scene class. Early researches propose to construct scene representations with variant low-level features, e.g., local structures [41], [42], color attributes [43], [44], and texture information [45], [46]. Concerning that low-level features fail to comprehensively depict complex scenes, mid-level algorithms, such as Bag-of-Visual-Words (BoVW) [47], [48] and topic models [49], [50], are devised to encode local features (so-called “visual words”) into more holistic mid-level scene representations for the classification task. However, these methods show limited performance in recognizing scenes of high diversity due to their dependency on hand-crafted features.

¹<https://earth.google.com/web/>

²<https://www.openstreetmap.org/>

TABLE I
COMPARISON WITH EXISTING AERIAL SCENE DATASETS FROM VARIOUS PERSPECTIVES.

Dataset	# images	spatial resolutions	# scenes	# labels per image	crowdsourced label	Year
UC-Merced [37]	2,100	0.3 m/pixel	21	1	✗	2010
WHU20 [38]	5,000	0.3-7.4 m/pixel	20	1	✗	2015
RSSCN7 [39]	2,800	0.2-1.4 m/pixel	7	1	✗	2015
AID [27]	10,000	0.5-8 m/pixel	30	1	✗	2017
NWPU-RESISC45 [40]	31,500	0.2-30 m/pixel	45	1	✗	2017
MultiScene (Ours)	100,000	0.3-0.6 m/pixel	36	1-13	✓	2021

Recently, the emergence of deep CNNs brings immense advancements to the community, and many achievements [19]–[34] have been obtained in the field of aerial single-scene classification. These deep networks have hierarchical architectures, where convolutional and max-pooling layers are periodically interleaved for learning high-level features of intricate scenes. With layers going deeper, the learned features are more abstract and supposed to contain richer semantic information, which is crucial for judicious decisions. A popular trend of deep learning algorithms in single-scene classification is to take a CNN as the backbone and introduce well-designed modules for further enhancing the feature efficiency. For instance, Bi et al. [31] propose to learn multiple instances from feature maps extracted by a densely-connected CNN and integrate them into bag-level features for single-scene classification. Li et al. [51] propose a key region capturing method to learn class-specific features and retain global information for inferring scene labels. To leverage features of variant levels, feature aggregation plays a key role in single-scene classification. Lu et al. [52] fuses features learned by the last three blocks and the second fully-connected layer of VGG-16, and Cao et al. [53] designs a non-parametric self-attention layer to enhance spatial and channel responses of fused features for the final prediction. In [20], the authors develop a gated bidirectional network for aggregating features extracted by different convolutional layers with a gated function in both top-down and bottom-up directions. Besides, exploiting supplementary data, such as geo-tagged audios and multi-temporal images, has been a new research direction. Hu et al. [19] propose to predict scene categories by transferring sound event knowledge learned from sound-image pairs. In [25], the authors propose a two-branch network to learn deep features of bi-temporal images and fuse them through a CorrFusion module for aerial scene classification. Our literature review demonstrates that most of the existing researches assume that an aerial image includes only one scene and focus on well-cropped single-scene aerial images. Hence, these studies tend to regard entities present in an image as compositions of a scene, while in multi-scene recognition, this would trigger networks to learn erroneous feature representations. However, very few efforts have been deployed to explore multi-scene recognition in the remote sensing community.

B. Multi-label Object Classification

Multi-label object classification refers to assigning an aerial image multiple object-level labels, such as car, tree, and building. Similar to our work, these studies aim to provide

a holistic understanding of aerial images, but from the perspective of object. Early attempts [54], [55] follow the idea of simply combining a deep CNN with a post-processing approach for identifying multiple objects in an aerial image. In [54], the authors feed outputs of a CNN into a customized thresholding operation for inferring multiple object labels, while in [55], a conditional random field (CRF) is utilized as the post-processing model. In recent literature, more efforts are deployed to endow deep neural networks with the capacity of reasoning about relations among various objects for more accurate predictions. In [56], the authors propose an end-to-end network comprising a CNN and a long short-term memory (LSTM) network that is responsible for modeling label dependencies through its recurrent units for multi-label object classification. [57] exploits a bidirectional LSTM network to learn spatial relations among all patches in an image for the final prediction. In [58], the authors propose a relational reasoning network module to model label dependencies and gains better classification results. Instead of encoding label relations, [59] divides an aerial image into several patches with the same size and models spatial relationships among them for multi-label object interpretation. Compared to these researches, our task is more challenging, because compared to object, the concept of scene is more abstract and intricate.

III. MULTISCENE DATASET FOR MULTI-SCENE RECOGNITION IN SINGLE AERIAL IMAGES

This section first reviews existing single-scene aerial image datasets and then delineates the proposed dataset.

A. Existing Single-scene Aerial Image Dataset

During the last decades, various aerial image datasets are published for single-scene classification, and here we briefly review several commonly used ones.

- *UC-Merced [37]*: The UC-Merced dataset is composed of 2,100 images collected from the United States Geological Survey (USGS) National Map, and each of them is categorized into one of 21 scene classes: overpass, golf course, river, harbor, beach, building, airplane, freeway, intersection, medium residential, runway, agricultural, storage tank, parking lot, forest, sparse residential, chaparral, tennis courts, dense residential, baseball diamond, and mobile home park. The number of images per scene is evenly defined as 100, and only cities in the United States are covered in data acquisition. The size of each image is 256×256 pixels, and the spatial resolution is one foot. In [60], the authors focus on the task of recognizing

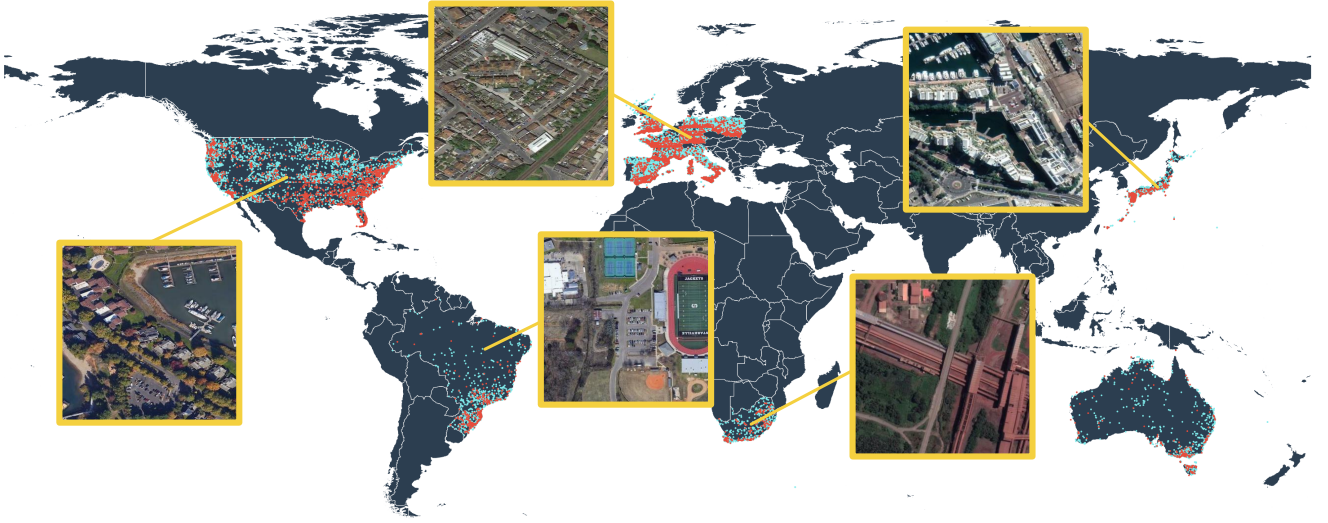


Fig. 3. Coordinate distributions and examples of multi-scene aerial images in our dataset. Red dots denote images with both crowdsourced and clean labels, and cyan dots represent images with only crowdsourced scene labels.

multiple objects in an image and relabel the UC-Merced dataset, yielding a multi-label dataset. In this dataset, 2,100 images are relabeled, and each is assigned one or several labels from 17 newly defined object classes: airplane, sand, pavement, building, car, chaparral, court, tree, dock, tank, water, grass, mobile home, ship, bare soil, sea, and field.

- **WHU20 [38]:** The WHU20 dataset is an extended version of the WHU-RS dataset that was originally proposed in [61]. This dataset expands numbers of aerial images and scene classes from 950 to 5,000 and from 12 to 20, respectively. For each scene category, more than 200 images with a size of 600×600 pixels are collected, and their spatial resolutions range from 0.26 m/pixel to 7.44 m/pixel.
- **RSSCN7 [39]:** The RSSCN7 dataset is a collection of 2,800 high-resolution images each belonging to one of 7 scene categories: grassland, forest, farmland, parking lot, river/lake, industrial region, and residential region. 400 images with different spatial resolutions are cropped from Google Earth imagery for each scene, and the image size is 400×400 pixels.
- **AID [27]:** The AID dataset is a large-scale benchmark consisting of 10,000 aerial images and 30 scene types: airport, pond, forest, baseball field, resort, bare land, center, beach, bridge, commercial, desert, storage tanks, farmland, industrial, mountain, park, parking, playground, viaduct, church, railway station, river, school, meadow, sparse residential, dense residential, medium residential, square, stadium, and port. Google Earth is exploited to acquire image samples, and the spatial resolution of each sample varies from 0.5 m/pixel to 8 m/pixel. The size of images is 600×600 pixels, and the number of images for each class ranges from 220 to 420.
- **NWPU-RESISC45 [40]:** The NWPU-RESISC45 dataset contains 31,500 high-resolution images and each is as-

signed with one of 45 scene labels. For each scene, 700 images with a size of 256×256 pixels are acquired from Google Earth imagery, and their spatial resolutions vary from 0.2 m/pixel to 30 m/pixel.

In addition, we note that BigEarthNet [62] is a large-scale dataset for multi-label learning, where 590,326 Sentinel-2 images are captured over the European Union, and their spatial resolutions range from 10 m/pixel to 60 m/pixel. Since BigEarthNet focuses on land covers instead of scenes, we do not specify it here. Table I presents an overview of public high-resolution aerial image datasets from the perspectives of dataset scales, image resolutions, scene categories, and annotations.

B. MultiScene for Multi-scene Recognition

Although there are already variant datasets for aerial scene recognition, most of them can only be used for single-scene classification. In this paper, we aim to take a step towards a more general scenario, multi-scene recognition in single images, and produce the MultiScene dataset.

To be more specific, we collect 100,000 high-resolution aerial images from Google Earth imagery, which cover six continents, Europe, Asia, North America, South America, Africa, and Oceania, and eleven countries including Germany, France, Italy, England, Spain, Poland, Japan, the United States, Brazil, South Africa, and Australia (cf. Fig. 3). This can ensure high intra-class diversity, as different scene appearances resulted from different cultural regions are covered. The spatial resolution of each image ranges from 0.3 m/pixel to 0.6 m/pixel, and the spatial size of images is 512×512 pixels. In contrast to single-scene image datasets [27], [37]–[39], we put no constraints on the location and area of the dominant/trivial scene in an image during the data collection process. Some example multi-scene images are exhibited in Fig. 4. In total, 36 scene categories are defined: apron, baseball field, basketball field, beach, bridge, cemetery, commercial, farmland,

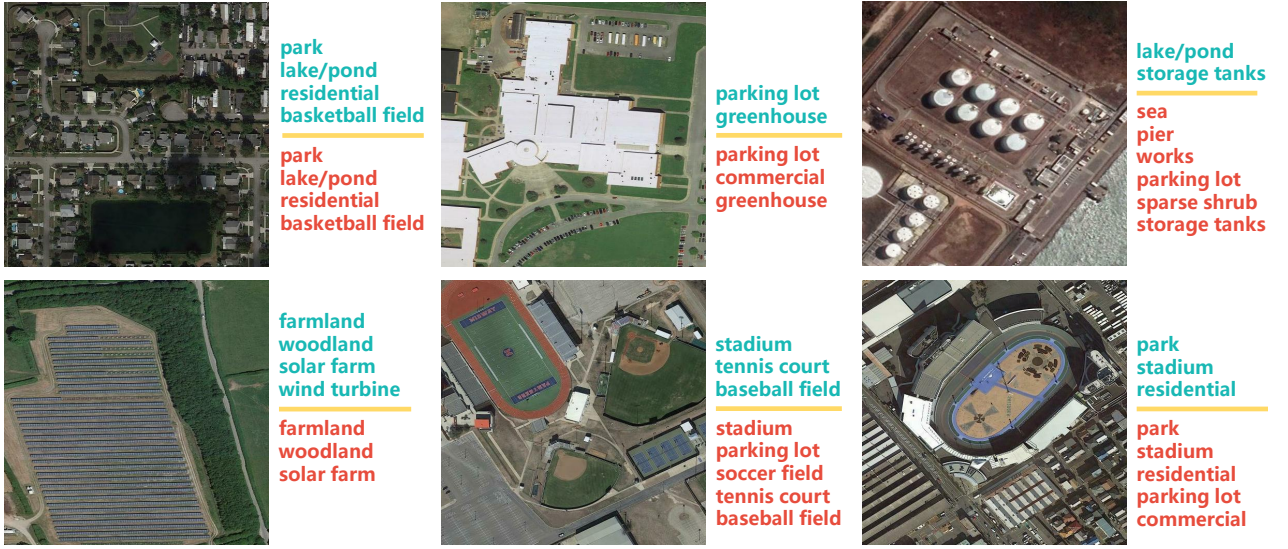


Fig. 4. Example multi-scene aerial images with their crowdsourced and clean annotations in the MultiScene dataset.

woodland, golf course, greenhouse, helipad, lake/pond, oil field, orchard, parking lot, park, pier, port, quarry, railway, residential, river, roundabout, runway, soccer field, solar farm, sparse shrub, stadium, storage tanks, tennis court, train station, wastewater, plant, wind turbine, works, and sea.

To obtain crowdsourced annotations, we first localize each image in OSM with coordinates of its four corners. Afterwards, we parse properties of scenes present in the corresponding region and label images accordingly. In this way, crowdsourced annotations of all aerial images can be automatically yielded at a very low cost compared to conventional manual labeling. However, these almost free annotations might suffer from noise as aforementioned in Section I, and the performance of networks directly trained on them could be degraded. Therefore, we visually inspect 14,000 images from all six continents and correct their labels, yielding a subset, MultiScene-Clean. Fig. 3 shows the coordinate distribution of all images, and the number of samples associated with each scene is present in Fig. 5. Compared to other scene recognition datasets (cf. Table I), our dataset is featured by its manifold labels per image and the available crowdsourced annotations. Fig. 6 further shows the number of images associated with different numbers of scenes.

C. Challenges

Compared to existing aerial scene datasets, our dataset brings more challenges to the field of scene interpretation from the following three perspectives:

- Images are unconstrained and large-scale, and thus scenes are likely to be incomplete and trivial, which makes recognition more difficult.
- The long-tail sample distribution (see Fig. 5) poses a challenge of learning unbiased models on an imbalanced dataset.
- We gather images from different cultural regions, which results in a high intra-class variation.

IV. EXPERIMENTS

A. Experimental Setup

Data Configuration. We evaluate the performance of existing models on both MultiScene-Clean and MultiScene datasets. As to the former, we use 7,000 cleanly labeled images to train and validate networks, and the remaining images are utilized to test networks. For the latter, we leverage the same test set but train deep neural networks on the other 93,000 images with only crowdsourced annotations.

Evaluation. For a comprehensive evaluation, we measure the performance of baseline models with class-based, example-based, and overall metrics. Let L and N be numbers of classes and examples³, these metrics are calculated as follows.

- **Class-based Metrics:** Mean class-based precision (mCP), recall (mCR), F_1 (mCF₁) score, and per-class average precision (AP) are calculated for measuring the performance of networks from the perspective of class. Specifically, mCP, mCR, and mCF₁ score are computed as:

$$\begin{aligned} \text{mCP} &= \frac{1}{L} \sum_{c=1}^L \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c}, \quad \text{mCR} = \frac{1}{L} \sum_{c=1}^L \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c}, \\ \text{mCF}_1 &= \frac{1}{L} \sum_{c=1}^L \frac{\text{TP}_c}{\text{TP}_c + \frac{1}{2}(\text{FP}_c + \text{FN}_c)}, \end{aligned} \quad (1)$$

where TP_c , FN_c , and FP_c represent numbers of true positives, false negatives, and false positives with respect to the c -th class, respectively. As to the per-class AP, we first rank all examples according to the predicted probability of the c -th class in each of them. Then

³An example indicates an image which has multiple labels.

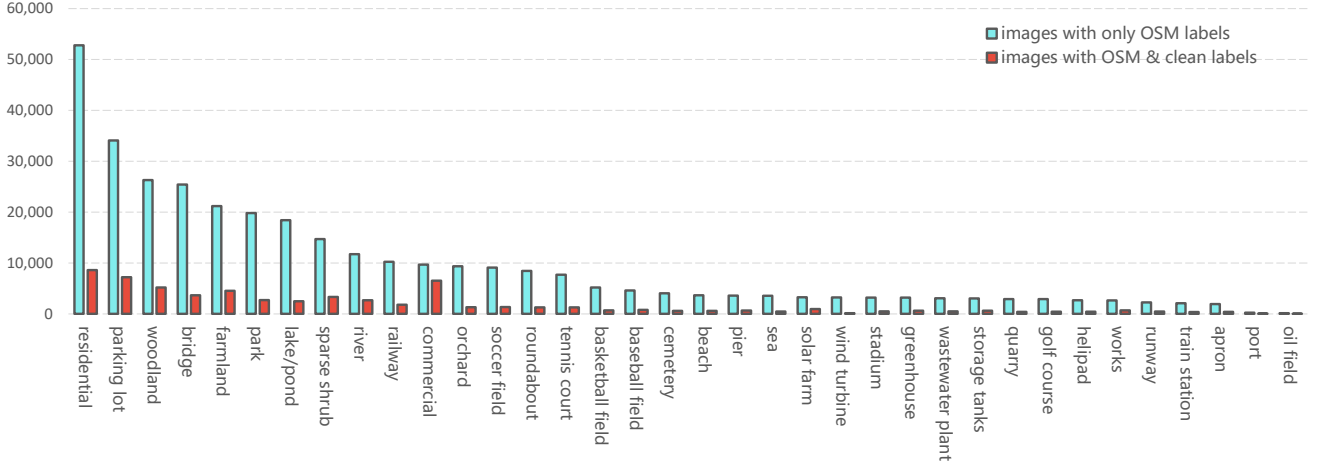


Fig. 5. Sample distributions of all scene categories in our dataset. Each cyan bar indicates the number of images assigned only OSM labels with respect to each scene category, and red bars represent numbers of images with both OSM and clean labels.

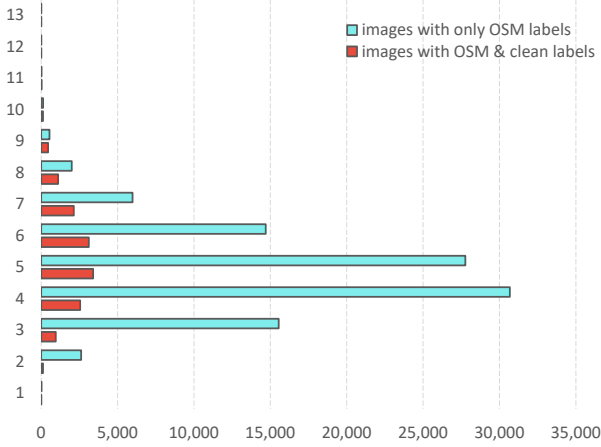


Fig. 6. The number of images associated with different numbers of scenes. Y-axis indicates the number of scenes, and X-axis represents the number of images. The legend is the same as that in Fig. 5.

we calculate the corresponding AP with the following formula:

$$AP = \frac{1}{N_c} \sum_{k=1}^N \frac{TP_c@k}{TP_c@k + FP_c@k} \times \text{rel}@k, \quad (2)$$

where N_c denotes the number of examples including the c -th class, and $TP_c@k$ and $FP_c@k$ represent numbers of true and false positives in top- k examples, respectively. Notably, $TP_c@k$ and $FP_c@k$ are equivalent to TP_c and FP_c , when k equals to N . $\text{rel}@k$ denotes the relevance between the k -th example and the c -th class, and it is set to 0/1 when the c -th class is included/excluded. Besides, the mean average precision (mAP) can be computed by averaging APs for all categories.

- **Example-based Metrics:** Mean example-based precision (mEP), recall (mER), and F_1 (mEF₁) score are computed to validate networks from the perspective of example with

the following equations:

$$\begin{aligned} mEP &= \frac{1}{N} \sum_{k=1}^N \frac{TP_k}{TP_k + FP_k}, \quad mER = \frac{1}{N} \sum_{k=1}^N \frac{TP_k}{TP_k + FN_k}, \\ mEF_1 &= \frac{1}{N} \sum_{k=1}^N \frac{TP_k}{TP_k + \frac{1}{2}(FP_k + FN_k)}, \end{aligned} \quad (3)$$

where TP_k , FP_k , and FN_k denote numbers of true positives, false positives, and false negatives in the k -th example.

- **Overall Metrics:** Overall precision (OP), recall (OR), and F_1 (OF₁) score can be used to measure the performance of models from a more holistic perspective, and they are calculated as:

$$\begin{aligned} OP &= \frac{TP}{TP + FP}, \quad OR = \frac{TP}{TP + FN}, \\ OF_1 &= \frac{TP}{TP + \frac{1}{2}(FP + FN)}, \end{aligned} \quad (4)$$

where TP, FP, and FN are counted based on predictions of all scenes and examples.

B. Baselines

To provide comprehensive benchmarks, we evaluate the performance of extensive popular deep neural networks. Since they were originally designed for single-label classification, we substitute sigmoid functions for their softmax activations to predict multiple scene labels that are encoded into multi-hot binary sequences. Besides, several classical machine learning algorithms are also evaluated. In total, 22 models are tested on both MultiScene-Clean and MultiScene datasets, and a brief review is as follows.

- **SVM [63]:** Support vector machine (SVM) aims to learn one or several hyperplanes for separating samples of different classes with the largest margin. Usually, the hyperplanes are constructed in a high dimensional space, and can be learned directly (Linear SVM) or through

TABLE II
NUMERICAL RESULTS OF BASELINE MODELS ON THE MULTISCENE-CLEAN DATASET (%). MODELS ARE TRAINED AND TESTED ON CLEANLY-LABELLED IMAGES, AND THE BEST SCORES ARE SHOWN IN BOLD.

Model	mAP	mCP	mCR	mCF ₁	mEP	mER	mEF ₁	OP	OR	OF ₁
SVM	14.9	19.6	8.4	8.6	62.2	32.8	41.1	66.9	32.2	43.5
RF	15.6	25.4	8.7	9.5	64.6	32.5	41.4	70.9	32.1	44.2
XGBOOST	16.9	34.1	11.2	12.8	67.0	37.4	45.8	69.6	36.5	47.9
VGG-16	56.5	63.3	47.9	53.6	74.9	64.3	67.0	73.6	63.1	67.9
VGG-19	56.4	62.9	47.7	53.3	74.8	64.1	66.8	73.5	62.7	67.7
Inception-V3	53.5	65.0	40.8	48.5	74.2	59.9	63.9	73.0	58.6	65.0
ResNet-50	62.0	74.8	45.9	55.1	79.7	62.7	67.9	79.0	61.4	69.1
ResNet-101	63.0	75.9	46.6	55.8	79.9	64.3	69.1	79.2	63.1	70.3
ResNet-152	63.8	74.9	49.1	57.7	80.8	64.0	69.2	80.1	62.8	70.4
SqueezeNet	46.3	58.1	36.8	43.5	71.3	58.0	61.3	70.0	56.9	62.7
MobileNet-V2	58.8	70.9	44.8	53.1	77.6	62.7	67.0	76.6	61.6	68.3
ShuffleNet-V2	50.7	61.8	38.1	45.7	73.8	58.2	62.5	73.0	57.0	64.0
DenseNet-121	62.2	74.6	45.1	54.4	79.5	61.8	67.3	79.1	60.6	68.6
DenseNet-169	63.2	76.7	45.8	55.3	80.4	63.4	68.6	79.6	62.3	69.9
ResNeXt-50	63.4	77.3	45.0	54.2	78.5	64.3	68.6	77.8	63.2	69.8
ResNeXt-101	64.8	76.5	48.6	57.3	79.3	66.6	70.2	78.5	65.4	71.3
MnasNet	53.8	61.8	42.9	49.9	73.0	59.4	63.0	72.1	58.1	64.3
KFBNNet	58.8	68.8	45.2	53.3	77.9	64.2	68.1	77.3	63.0	69.4
FACNN	56.5	60.3	48.7	52.6	73.1	65.3	66.8	71.6	64.1	67.7
SAFF	61.8	72.5	48.1	56.7	79.4	63.9	68.6	78.7	62.8	69.9
LR-VGG-16	58.1	67.7	46.7	54.2	77.3	64.6	68.0	76.2	63.5	69.2
LR-ResNet-50	63.1	68.1	53.1	59.0	76.7	67.6	69.7	75.3	66.5	70.6

kernel functions (Nonlinear SVM). In our experiments, we select the latter and use a radial basis function (RBF) kernel [64] to learn SVM.

- *RF* [65]: Random forest (RF) is an ensemble of decision trees, which are trained with random subspaces of image features and make final predictions through the majority voting. The number of decision trees is set to 200 in our experiments.
- *XGBOOST*: XGBOOST⁴ is a computationally efficient implementation of gradient-boosted trees [66] that optimizes tree ensembles (e.g., an ensemble of decision trees) through successive learning steps [67]. In each step, the existing trees are fixed, and a new tree is added and optimized with objective functions. Considering the difficulty of our task, we set the number of trees to 200 for training XGBOOST on both datasets.
- *VGGNet* [68]: VGGNet utilizes five convolutional blocks and three fully-connected layers to extract high-level features for image classification. Each block has multiple stacked convolutional layers and ends with one max-pooling layer. The size of convolutional filters is 3×3 , and the stride of max-pooling layers is 2. In our experiments, a 16-layer VGGNet (VGG-16) and a 19-layer VGGNet (VGG-19) are trained on our dataset.
- *Inception networks* [69]–[72]: Inception networks are characterized by their wide modules, where convolutional filters of variant sizes and max-pooling operators are jointly employed to learn diverse features. Besides, a bottleneck architecture made of 1×1 convolutions is introduced to mitigate the boosted computational cost resulting from heavy inception modules. In Table II and III, we report the performance of Inception-v3 [71] in multi-scene recognition.

- *ResNet* [73]: ResNet aims to address the degradation problem by learning residual mappings with shortcut connections. By doing so, ResNet can go much deeper than plain CNNs and achieve outstanding performance in not only image classification but also semantic segmentation and object detection tasks. In our experiments, we evaluate a 50-layer ResNet (ResNet-50), a 101-layer ResNet (ResNet-101), and a 152-layer ResNet (ResNet-152) on the proposed dataset. Notably, residual blocks in these deep ResNets are modified into bottleneck architectures for reducing the computational burden.
- *SqueezeNet* [74]: SqueezeNet focuses on preserving network performance with fewer parameters. To achieve this, most of 3×3 convolutional filters are replaced with 1×1 filters, and features are squeezed in the channel dimension before fed into the remaining 3×3 filters. In addition, bypass connections are introduced to features of the same size for improving the classification performance. Experimental results of SqueezeNet on our dataset are reported in Section IV-D and IV-D2.
- *MobileNet* [75]: MobileNet is a light-weight deep neural network, which is applicable on mobile devices with restricted computational sources. The network is designed in a streamlined architecture, and depthwise separable convolutions play a significant role in increasing computational efficiency. Specifically, such convolutions are implemented by factorizing standard convolutions into depthwise and pointwise convolutions. The former is conducted on each channel, and the latter aggregates channel-wise outputs via 1×1 convolutions. To further reduce the computational cost, two hyperparameters, width multiplier α and resolution multiplier β , are designed to shrink feature channels and input resolutions, respectively. In the advanced variation of MobileNet, i.e., MobileNet-V2 [76], inverted residual connections and

⁴<https://xgboost.readthedocs.io/en/latest/tutorials/model.html>

TABLE III
COMPARISONS OF APs ON THE MULTISCENE-CLEAN DATASET (%). THE BEST APs ARE SHOWN IN BOLD.

Model	apron	baseball field	beach	bridge	cemetery	farmland	woodland	golf course	greenhouse	lakepond	oil field	orchard	parking lot	park	pier	port	quarry	railway	residential	roundabout	soccer field	sparse shrub	stadium	storage tanks	tennis court	train station	wastewater plant	wind turbine	works	sea						
SVM	3.1	6.3	5.4	4.0	29.9	4.5	60.5	44.1	55.2	3.1	5.2	3.0	18.0	0.1	9.9	65.5	20.2	5.5	0.6	3.0	13.4	68.3	19.3	8.7	3.7	9.5	8.0	25.7	3.7	5.2	9.1	2.7	3.6	0.8	5.2	3.4
RF	3.1	6.3	5.4	10.9	28.1	4.5	62.7	53.2	54.6	3.1	5.4	3.0	17.4	0.1	10.1	65.6	20.2	7.1	0.6	3.0	13.6	73.3	19.5	8.7	3.7	9.5	10.6	26.2	3.7	4.9	9.1	2.7	3.6	0.8	5.2	3.4
XGBOOST	3.1	9.4	5.4	15.8	33.4	4.5	62.8	56.2	57.6	3.1	5.2	3.0	21.5	0.1	11.5	68.8	23.3	9.3	0.6	3.0	13.8	74.7	20.3	8.7	4.2	9.8	10.9	32.9	3.7	5.2	9.1	2.7	3.6	0.8	5.2	4.3
VGG-16	72.2	81.7	24.2	70.0	72.1	28.9	81.6	87.8	85.7	65.1	42.8	34.9	63.2	1.9	72.0	86.2	50.1	72.7	19.0	50.9	55.6	93.7	52.5	65.8	68.7	61.6	32.4	59.7	53.7	49.4	63.6	35.5	45.9	51.6	27.2	54.9
VGG-19	70.1	80.7	21.3	67.1	71.7	28.0	80.6	87.4	85.5	64.5	44.4	33.5	64.1	2.6	73.4	86.6	50.7	72.4	17.2	50.7	56.4	93.8	52.3	68.7	68.6	62.3	32.3	58.2	53.0	49.1	66.8	37.6	47.0	50.3	28.3	54.5
Inception-V3	67.8	83.2	20.0	68.8	69.9	19.8	81.4	85.0	83.1	51.4	36.4	33.0	55.2	0.5	68.3	86.0	50.0	70.5	13.5	49.0	47.9	92.3	49.8	63.9	65.4	59.7	31.1	56.6	57.9	44.6	55.5	34.5	46.9	44.5	22.7	59.6
ResNet-50	77.3	86.7	26.4	79.4	74.6	39.7	83.3	88.4	86.7	76.7	49.3	43.1	66.0	0.5	76.8	88.2	55.3	77.2	24.7	55.7	62.4	94.3	59.6	71.2	74.8	67.7	40.3	61.9	63.0	55.0	68.8	46.7	54.5	50.3	36.0	68.9
ResNet-101	79.7	88.0	27.6	80.2	75.9	44.5	84.2	88.4	87.3	75.6	49.7	45.3	68.7	0.9	77.8	88.6	58.3	77.6	20.9	61.2	62.7	94.5	61.5	73.3	77.4	70.0	40.0	62.1	64.6	54.2	70.9	46.6	55.8	47.2	37.8	68.1
ResNet-152	79.1	88.6	27.4	84.0	77.1	42.4	83.9	88.7	87.6	77.4	51.8	46.9	68.8	0.4	78.3	89.2	59.4	79.3	20.3	59.4	65.1	94.5	61.9	74.4	77.7	70.7	41.0	62.8	65.1	57.2	72.5	51.3	57.3	48.8	35.5	71.1
SqueezeNet	51.4	73.8	18.2	57.6	59.4	15.6	77.4	83.8	81.6	50.1	34.0	12.3	51.3	0.7	65.9	83.7	44.2	56.8	20.5	34.7	46.2	93.2	44.9	57.7	58.0	45.7	27.4	53.2	38.9	35.9	56.1	22.8	26.8	18.0	18.9	49.1
MobileNet-V2	74.3	84.4	24.8	78.5	73.4	32.5	81.9	87.2	85.9	72.2	46.3	38.9	64.3	1.6	73.8	88.3	53.7	72.2	18.1	51.8	60.0	93.6	55.8	71.7	67.8	64.1	34.3	60.4	58.7	46.3	69.8	40.7	47.1	48.5	31.5	64.0
ShuffleNet-V2	62.0	78.4	23.5	67.5	66.8	14.4	80.5	84.1	82.5	61.0	36.6	18.0	56.5	0.4	65.0	85.5	48.6	61.9	11.0	40.4	50.6	92.5	48.9	57.0	66.5	59.4	31.0	57.3	51.7	37.5	56.3	30.2	36.4	20.4	27.0	58.8
DenseNet-121	79.0	87.5	28.3	80.9	75.1	37.9	83.5	87.7	85.5	78.0	48.0	47.3	66.3	2.9	75.7	88.5	58.3	77.7	25.3	58.4	62.3	94.1	58.5	73.6	73.9	67.1	39.1	61.6	61.4	54.3	69.9	47.4	56.4	50.2	31.0	66.5
DenseNet-169	81.8	88.1	26.3	81.4	76.9	42.6	84.5	88.1	86.0	77.2	49.0	44.1	67.4	4.3	78.7	88.8	56.6	78.1	21.4	58.2	62.9	94.4	60.5	72.4	75.4	69.4	41.2	61.2	65.9	54.6	72.4	51.4	58.6	53.2	35.1	67.6
ResNeXt-50	81.5	87.1	27.6	81.6	75.5	41.3	83.4	88.2	86.3	76.1	50.8	50.3	66.3	9.6	75.3	89.0	57.6	77.7	23.8	58.6	64.5	94.1	61.4	73.3	76.7	68.7	40.9	62.2	63.5	53.8	71.8	50.2	56.9	54.7	34.9	66.6
ResNeXt-101	82.3	87.7	30.2	82.9	77.2	45.6	84.1	88.8	87.3	77.1	52.6	54.8	71.0	1.3	79.4	89.7	58.6	76.1	20.3	61.8	66.5	94.5	64.3	75.0	79.2	72.0	42.3	63.3	64.5	55.4	75.1	54.4	58.7	52.5	37.8	67.2
MnasNet	69.4	84.0	21.6	70.9	67.4	20.7	78.9	84.7	82.4	63.0	42.0	27.5	58.3	0.7	70.5	85.7	49.1	69.5	18.1	45.2	51.6	91.4	48.0	66.1	66.4	59.0	33.3	55.9	53.2	42.4	61.5	32.8	42.6	33.4	25.6	62.6
KFBNNet	68.4	83.0	27.2	75.1	75.4	37.6	82.2	88.7	86.7	68.4	47.8	47.0	67.4	6.3	75.7	89.1	55.3	73.1	10.1	57.2	58.5	94.6	55.1	72.5	68.7	64.7	35.4	60.3	46.5	48.5	74.8	31.6	47.3	49.1	26.0	61.2
FACNN	69.4	83.7	21.5	70.5	72.9	31.7	81.8	88.4	85.4	66.4	36.9	36.2	65.9	4.8	72.8	87.5	50.4	71.1	12.6	57.8	54.5	93.6	55.2	70.1	68.4	64.5	32.4	56.4	50.0	50.8	68.4	32.3	46.1	46.4	23.9	54.2
SAFF	74.5	86.2	29.8	76.8	75.1	41.4	83.0	88.7	86.6	76.9	50.0	49.6	67.7	1.9	76.8	88.9	58.7	74.8	17.1	51.6	62.1	94.2	58.8	76.5	72.4	68.3	39.8	63.3	57.6	54.5	77.8	41.9	54.6	46.6	33.7	65.9
LR-VGG-16	76.3	82.5	19.9	74.7	71.0	26.6	82.5	86.8	86.4	70.7	41.4	41.0	65.1	11.0	72.0	87.5	52.5	73.1	10.4	57.4	58.5	93.5	56.3	70.7	71.2	62.7	27.3	58.1	51.6	50.5	69.0	43.9	52.0	45.9	30.5	62.5
LR-ResNet-50	78.5	88.2	24.6	80.8	75.9	44.2	83.8	88.7	87.8	76.2	50.9	48.1	67.4	0.9	77.1	88.8	57.3	76.8	29.3	57.5	63.3	94.2	61.0	72.4	73.0	70.5	42.1	62.3	64.8	55.1	72.0	48.7	55.9	48.9	36.4	67.3

TABLE IV
EXAMPLE PREDICTIONS OF RESNEXT-101 ON THE MULTISCENE-CLEAN DATASET.

Multi-scene Aerial Images in the MultiScene-Clean dataset					
Ground Truths	bridge, parking lot, river, roundabout, and residential	woodland, lake/pond, and wastewater plant	bridge, parking lot, river, roundabout, and residential	farmland, woodland, orchard, residential, and sparse shrub	lake/pond and quarry
Predictions	bridge, parking lot, river, roundabout, and residential	woodland, lake/pond, and wastewater plant	bridge, parking lot, river, roundabout, and residential	farmland, woodland, orchard, residential, and sparse shrub	lake/pond and quarry
Multi-scene Aerial Images in the MultiScene-Clean dataset					
Ground Truths	commercial, farmland, parking lot, and residential	commercial, parking lot, park, railway, residential, train station, and works	farmland, woodland, sparse shrub	baseball field, basketball field, lake/pond, parking lot, residential, soccer field, and tennis court	bridge, commercial, parking lot, park, residential, river, roundabout, and solar farm
Predictions	commercial, farmland, woodland , parking lot , and residential	commercial, parking lot, park , railway, residential, train station, and works	farmland, woodland, lake/pond , and sparse shrub	baseball field , baseball field , lake/pond, parking lot , residential, soccer field , and tennis court	bridge, commercial, lake/pond , parking lot, park , river , solar farm , and residential

Purple predictions indicate false negatives, while blue predictions are false positives.

linear bottlenecks are developed to improve the network performance. In our experiments, we train MobileNet-V2 and set both α and β as the default value, 1.

- *ShuffleNet* [77]: ShuffleNet improves computational efficiency by utilizing pointwise group convolutions and channel shuffle. Specifically, the former divides feature maps into several groups and conducts 1×1 convolutions on each group independently. The latter rearranges feature channels for enabling information to flow across channels belonging to different groups. Besides, element-wise addition, which is often used in a residual block, is replaced with concatenation for enlarging channel dimension at a low computational cost. In ShuffleNet-V2 [78], features are grouped by channel split, and pointwise group convolutions are discarded. As a consequence, two feature groups are yielded and fed into two branches, of which one is an identity mapping and the other is a set of convolutions. Afterwards, outputs are concatenated and shuffled along the channel dimension. In our experiments, we evaluate the performance of ShuffleNet-V2 on our dataset.
- *DenseNet* [79]: DenseNet proposes to enhance information flow by directly connecting each layer to all subsequent layers with equivalent feature-map sizes. To

preserve information learned by proceeding layers, concatenation is employed to combine features from various layers. By reusing feature maps throughout entire networks, DenseNet can learn compact internal representations for visual recognition tasks. Two variations, a 121-layer DenseNet (DenseNet-121) and a 169-layer DenseNet (DenseNet-169), are tested.

- *ResNeXt* [80]: ResNeXt learns residuals with aggregated residual transformations but not a stack of convolutional layers (e.g., ResNet). The aggregated residual transformation is implemented by first slicing features into low-dimensional embeddings and then conducting convolutions on them. Afterwards, outputs are aggregated with element-wise addition. With this design, ResNeXt outperforms its ResNet counterpart on ImageNet-5K [80] and COCO [81] datasets. We test a 50-layer ResNext (ResNeXt-50) and a 101-layer (ResNeXt-101) in our experiments.
- *MnasNet* [82]: MnasNet architectures are automatically learned on target datasets through a mobile neural architecture search (MNAS) algorithm [82]. Compared to conventional NAS algorithms [83], MNAS takes not only classification accuracy but also model latency into consideration and is executed on mobile phones for measuring

TABLE V
NUMERICAL RESULTS OF BASELINE MODELS ON THE MULTISCENE DATASET (%). MODELS ARE TRAINED ON IMAGES WITH NOISY CROWDSOURCED ANNOTATIONS AND TESTED ON CLEANLY-LABELED IMAGES. THE BEST SCORES ARE SHOWN IN BOLD.

Model	mAP	mCP	mCR	mCF ₁	mEP	mER	mEF ₁	OP	OR	OF ₁
SVM	14.7	24.7	4.1	5.4	51.4	15.7	23.1	77.7	15.5	25.8
RF	15.1	49.7	4.4	6.1	55.4	16.4	34.3	78.7	15.8	26.3
XGBOOST	18.4	54.6	10.6	14.9	62.0	26.7	35.1	70.4	25.5	37.4
VGG-16	63.4	71.0	46.9	54.1	78.4	51.6	59.3	79.3	49.6	61.0
VGG-19	59.8	68.9	47.2	54.1	75.5	52.2	58.9	75.1	50.2	60.2
Inception-V3	65.8	74.1	50.8	58.5	79.1	53.8	61.2	79.5	51.9	62.8
ResNet-50	63.9	73.7	47.7	55.9	78.3	52.5	60.0	78.5	50.7	61.6
ResNet-101	63.4	73.0	47.5	55.5	77.1	52.5	59.7	77.2	50.6	61.2
ResNet-152	62.8	73.2	47.6	55.7	76.2	53.1	59.9	76.5	51.3	61.4
SqueezeNet	61.4	74.4	41.1	50.5	78.9	47.7	56.4	80.7	45.9	58.5
MobileNet-V2	65.5	72.3	48.4	56.0	79.6	54.6	62.0	80.1	52.8	63.6
ShuffleNet-V2	65.1	74.6	46.7	55.1	81.7	51.0	59.9	82.9	49.0	61.6
DenseNet-121	67.5	77.0	49.4	58.2	82.2	54.4	62.6	82.8	52.3	64.1
DenseNet-169	64.2	71.3	53.3	59.3	77.2	55.7	62.0	77.1	53.9	63.4
ResNeXt-50	63.9	73.6	49.0	56.9	77.5	52.6	59.8	77.6	50.7	61.3
ResNeXt-101	60.8	68.5	47.4	53.7	73.8	51.2	57.7	74.0	49.5	59.3
MnasNet	58.1	74.1	31.0	40.4	75.0	38.0	47.6	80.4	36.0	49.7
KFBNet	67.1	77.7	46.2	54.3	80.2	54.0	61.7	81.1	52.3	63.6
FACNN	65.2	73.9	47.6	55.6	78.9	53.9	61.1	80.0	52.1	63.1
SAFF	64.8	74.0	47.4	55.3	80.9	51.2	59.8	81.8	49.4	61.6
LR-VGG-16	67.8	76.0	48.4	56.1	80.5	52.1	60.5	81.3	50.3	62.2
LR-ResNet-50	65.5	71.2	51.6	57.9	79.2	53.1	60.7	79.4	51.2	62.3

real-world inference latency. As a consequence, MnasNet searched on target datasets is expected to achieve a good trade-off between accuracy and latency. To control the model size, a depth multiplier is designed for scaling the number of channels in each layer. In our experiments, the depth multiplier is set to 1, and the best-performing MnasNet searched on the ImageNet dataset [84] is chosen to perform multi-scene recognition in the wild.

- *KFBNet* [51]: KFBNet exploits a key region capturing method namely key filter bank (KFB) for aerial image scene classification. The proposed KFB is composed of two streams: a global stream (G-Stream) and a key stream (K-Stream). The former predicts labels using features learned by the last block of a CNN, while the latter highlights key features in both spatial and channel dimensions for inferring scene categories. Finally, predictions made by the two streams are merged via an element-wise addition as the final decision. We take VGG-16 as the backbone and report numerical results in Table II, III, and V.
- *FACNN* [52]: FACNN is a scene classification network composed of a CNN backbone and a Feature Aggregation module. In the latter, features extracted by the last three blocks of VGG-16 are aggregated through pooling operations and 1×1 convolutions. Afterwards, they are concatenated with outputs of the second fully-connected layer of VGG-16 to form discriminative scene representations for the final prediction.
- *SAFF* [53]: SAFF proposes a non-parametric self-attention layer for enhancing spatial and channel responses of feature maps. Specifically, features extracted by the last three blocks of a pre-trained CNN (e.g., VGG-16) are fused and fed into the proposed self-attention layer. In this layer, spatial- and channel-wise weightings are conducted to emphasize the importance of locations

of salient objects and channels with infrequently occurring features, respectively. Principal component analysis (PCA) whitening is also introduced to reduce the information redundancy and squash channels. However, since this operation frequently fails the network training, we replace it with a learnable fully-connected layer. Besides, VGG-16 is selected as the backbone in our experiments.

- *LR-CNN* [58]: LR-CNN is a multi-label classification network, which consists of three elements: a class-wise feature extraction module, an attentional region extraction module, and a relational reasoning module. Specifically, the first module learns deep features with respect to each category from the input images. Afterwards, the second module extracts attentional regions of class-wise features, which are eventually leveraged to reason about relations between different objects for inferring their existences through the third module. In our experiments, we validate LR-VGG-16 and LR-ResNet-50, where VGG-16 and ResNet-50 are taken as backbones, respectively.

C. Training Details

Before training SVM, RF, and XGBOOST, we use histogram of oriented gradient (HOG) [85] and local binary pattern (LBP) [86] as visual features as recommended in [87]. The size of each cell is set to 32×32 pixels for HOG, and the radius is defined as 16 pixels for LBP. We use Scipy to implement these machine learning classifiers and apply them to multi-scene recognition using the function *MultiOutputClassifier*⁵. As to baseline classification neural networks, we initialize them with weights pre-trained on the ImageNet dataset and fine-tune them on the proposed multi-scene image dataset. The loss is defined as binary cross-entropy, and stochastic

⁵<https://scikit-learn.org/stable/modules/generated/sklearn.multioutput.MultiOutputClassifier.html>

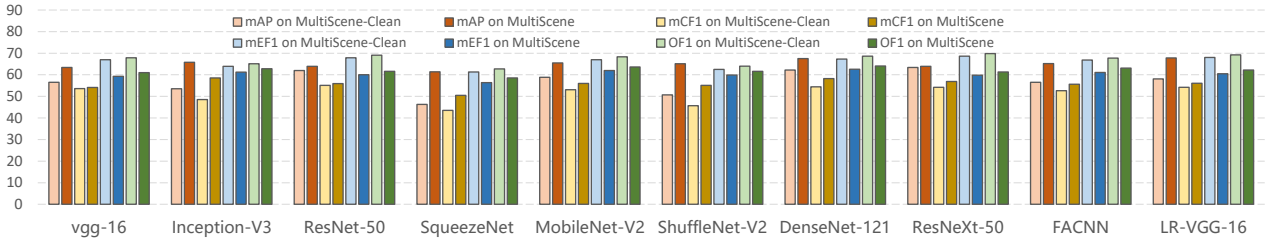


Fig. 7. Comparisons of the performance of networks trained on images with clean (light-color bars) and crowdsourced (dark-color bars) annotations, respectively. For each network, the left four bars represent class-based scores, mAPs and CF₁, while the right four bars indicate EF₁ and OF₁ scores.

gradient descent (SGD) with momentum [88] is selected as the optimizer. To accelerate the network convergence, the momentum is set to a large value, 0.9. Besides, the initial learning rate and weight decay are set to 0.02 and $1e-4$, respectively. All deep networks are implemented on Pytorch and validated on one NVIDIA Tesla V100-SXM2 32GB GPU. For experiments on both MultiScene-Clean and MultiScene, we train networks for 87k and 581k iterations, respectively, and the size of each training batch is set to 16 for both versions.

D. Experimental Results across Different Tasks

1) *Multi-scene Recognition with Cleanly-labeled Data*: To evaluate baselines for our task, we conduct experiments on the MultiScene-Clean dataset and report quantitative results in Table II. It can be seen that ResNeXt-101 achieves the best mAP (64.8%), mEF₁ (70.2%), and OF₁ score (71.3%), which demonstrate its high performance and robustness in this task from almost all perspectives. LR-ResNet-50 gains the highest value in mCF₁ (59.0%) owing to its capability of reasoning about relations among various scenes. Moreover, such a reasoning capability also enables LR-ResNet-50 to surpass the other baselines in all recall metrics, as scenes tend to be predicted as positive once its related scenes are recognized. Another observation is that MnasNet, SqueezeNet, and ShuffleNet-V2 show relatively poor performance due to their light-weight designs. Compared to deep neural networks, traditional machine learning algorithms achieve lower scores in all metrics.

For an insight into the performance of networks in identifying different scenes, we also report per-class APs in Table III. As we can see, ResNeXt-101 achieves the highest APs in most scenes, which is in line with the previous observations. Furthermore, we note that most networks fail to accurately recognize scenes having scarce training samples, e.g., oil field and port. This suggests that learning unbiased models on an imbalanced dataset is a big challenge. Besides numerical results, we exhibit several predictions in Table IV.

2) *Learning from Noisy Crowdsourced Labels*: We investigate networks learned from noisy crowdsourced labels for our task on the MultiScene dataset. To ensure a fair comparison, we utilize the same test set as in Section IV-D and report numerical results in Table V. It can be observed that OF₁ scores of all models are decreased by an average of 8.2% compared to the values in Table II, which demonstrates that noise in

crowdsourced annotations significantly affects the learning of deep neural networks. Moreover, it is interesting to note that the values of class-based metrics, mAP and mCF₁ score, are increased by 4.6% and 1.2%, respectively, in comparison with those in Table II. This can be attributed to the fact that numbers of training samples, especially for scenes seldomly appearing, are effortlessly increased by crawling OSM data with keyword searching. Compared to models showing high performance on the MultiScene-Clean dataset, we find that DenseNet gains the highest scores in mCF₁ (59.3%), mEF₁ (62.6%), and mOF₁ (64.1%), as it can sufficiently reuse features and has relatively few parameters. Besides, LR-VGG-16 achieves the highest mAP (67.8%), which demonstrates that taking advantage of underlying relations among various scenes can suppress the influence of noise introduced by OSM data. Furthermore, we compare the performance of several networks trained on MultiScene-Clean and MultiScene datasets in Fig. 7, and it can be again observed that higher class-based scores (see orange and brown bars in Fig. 7) are obtained when using massive crowdsourced labels. All in all, although crowdsourced labels influence the overall performance of networks, comparisons in class-based scores also suggest their great potential.

V. CONCLUSION

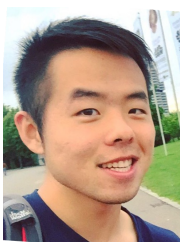
In this paper, we propose a large-scale dataset, MultiScene, for multi-scene recognition in single images, which is featured by unconstrained multi-scene aerial images and the available both crowdsourced and clean labels. The proposed dataset allows researches in not only recognizing aerial scenes in the wild but also learning from noisy crowdsourced labels. We comprehensively evaluate popular baseline models on both MultiScene-Clean (a subset consisting of only cleanly-labeled images) and MultiScene datasets. Experimental results on the former demonstrate that unconstrained multi-scene recognition is still a challenging task, and those on the latter showcase the great potential of exploiting a large number of crowdsourced annotations. Looking into the future, the dataset can be applied to develop more efficient networks and learning strategies for exploiting noisy labels for aerial scene understanding in the wild.

REFERENCES

- [1] Q. Weng, Z. Mao, J. Lin, and X. Liao, "Land-use scene classification based on a CNN using a constrained extreme learning machine," *International Journal of Remote Sensing*, pp. 1–19, 2018.

- [2] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1865–1883, 2017.
- [3] D. Wen, X. Huang, H. Liu, W. Liao, and L. Zhang, "Semantic classification of urban trees using very high resolution satellite imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 4, pp. 1413–1424, 2017.
- [4] S. Manfreda, M. McCabe, P. Miller, R. Lucas, V. Pajuelo Madrigal, G. Mallinis, E. Ben Dor, D. Helman, L. Estes, G. Ciraolo, J. Müllerová, F. Tauro, M. De Lima, L. De Lima, A. Maltese, F. Frances, K. Caylor, M. Kohv, M. Perks, G. Ruiz-Pérez, Z. Su, G. Vico, and B. Toth, "On the use of unmanned aerial systems for environmental monitoring," *Remote sensing*, vol. 10, no. 4, p. 641, 2018.
- [5] L. Mou and X. X. Zhu, "IM2HEIGHT: Height estimation from single monocular imagery via fully residual convolutional-deconvolutional network," *arXiv:1802.10249*, 2018.
- [6] C. Qiu, L. Mou, M. Schmitt, and X. X. Zhu, "Local climate zone-based urban land cover classification from multi-seasonal Sentinel-2 images with a recurrent residual network," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 154, pp. 151–162, 2019.
- [7] D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Datcu, and U. Stilla, "Classification with an edge: Improving semantic image segmentation with boundary detection," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 135, pp. 158–172, 2018.
- [8] N. Audebert, B. L. Saux, and S. Lefèvre, "Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 140, pp. 20–32, 2018.
- [9] L. Mou and X. X. Zhu, "RiFCN: Recurrent network in fully convolutional network for semantic segmentation of high resolution remote sensing images," *arXiv:1805.02091*, 2018.
- [10] Q. Li, L. Mou, Q. Liu, Y. Wang, and X. X. Zhu, "HSF-Net: Multi-scale deep feature embedding for ship detection in optical remote sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, 2017.
- [11] C. Qiu, M. Schmitt, C. Geiß, T. Chen, and X. X. Zhu, "A framework for large-scale mapping of human settlement extent from Sentinel-2 images via fully convolutional neural networks," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 163, pp. 152–170, 2020.
- [12] Q. Li, Y. Shi, X. Huang, and X. X. Zhu, "Building footprint generation by integrating convolution neural network with feature pairwise conditional random field (FPCRF)," *IEEE Transactions on Geoscience and Remote Sensing*, 2020.
- [13] G. Cheng, L. Guo, T. Zhao, J. Han, H. Li, and J. Fang, "Automatic landslide detection from remote-sensing imagery using a scene classification method based on BoVW and pLSA," *International Journal of Remote Sensing*, vol. 34, no. 1, pp. 45–59, 2013.
- [14] Q. Zhu, Y. Zhong, B. Zhao, G.-S. Xia, and L. Zhang, "Bag-of-visual-words scene classifier with local and global features for high spatial resolution remote sensing imagery," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 6, pp. 747–751, 2016.
- [15] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative cnns," *IEEE transactions on geoscience and remote sensing*, vol. 56, no. 5, pp. 2811–2821, 2018.
- [16] D. Marcos, M. Volpi, B. Kellenberger, and D. Tuia, "Land cover mapping at very high resolution with rotation equivariant CNNs: Towards small yet accurate models," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 145, pp. 96–107, 2018.
- [17] A. Vetrivel, M. Gerke, N. Kerle, F. Nex, and G. Vosselman, "Disaster damage detection through synergistic use of deep learning and 3D point cloud features derived from very high resolution oblique aerial images, and multiple-kernel-learning," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 140, pp. 45–59, 2018.
- [18] W. Lee, S. Kim, Y. Lee, H. Lee, and M. Choi, "Deep neural networks for wild fire detection with unmanned aerial vehicle," in *IEEE International Conference on Consumer Electronics (ICCE)*, 2017.
- [19] D. Hu, X. Li, L. Mou, P. Jin, D. Chen, L. Jing, X. X. Zhu, and D. Dou, "Cross-task transfer for multimodal aerial scene recognition," in *European Conference on Computer Vision (ECCV)*, 2020.
- [20] H. Sun, S. Li, X. Zheng, and X. Lu, "Remote sensing scene classification by gated bidirectional network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 1, pp. 82–96, 2020.
- [21] J. Murray, D. Marcos, and D. Tuia, "Zoom In, Zoom Out: Injecting scale invariance into landuse classification CNNs," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019.
- [22] G. Cheng, X. Xie, J. Han, L. Guo, and G. Xia, "Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 3735–3756, 2020.
- [23] A. Byju, G. Sumbul, B. Demir, and L. Bruzzone, "Remote sensing image scene classification with deep neural networks in JPEG 2000 compressed domain," *arXiv:2006.11529*, 2020.
- [24] Y. Xu, B. Du, and L. Zhang, "Assessing the threat of adversarial examples on deep neural networks for remote sensing scene classification: Attacks and defenses," *IEEE Transactions on Geoscience and Remote Sensing*, 2020.
- [25] L. Ru, B. Du, and C. Wu, "Multi-temporal scene classification and scene change detection with correlation based fusion," *arXiv:2006.02176*, 2020.
- [26] Q. Li, C. Qiu, L. Ma, M. Schmitt, and X. X. Zhu, "Mapping the land cover of Africa at 10 m resolution from multi-source remote sensing data with Google Earth engine," *Remote Sensing*, vol. 12, no. 4, p. 602, 2020.
- [27] G. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, and X. Lu, "AID: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, 2017.
- [28] P. Jin, G. Xia, F. Hu, Q. Lu, and L. Zhang, "AID++: An updated version of aid on scene classification," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2018.
- [29] D. Tuia, D. Marcos, and G. Camps-Valls, "Multi-temporal and multi-source remote sensing image classification by nonlinear relative normalization," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 120, pp. 1–12, 2016.
- [30] S. Niazmardi, B. Demir, L. Bruzzone, A. Safari, and S. Homayouni, "Multiple kernel learning for remote sensing image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 3, pp. 1425–1443, 2017.
- [31] Q. Bi, K. Qin, Z. Li, H. Zhang, K. Xu, and G. Xia, "A multiple-instance densely-connected ConvNet for aerial scene classification," *IEEE Transactions on Image Processing*, vol. 29, pp. 4911–4926, 2020.
- [32] J. Lin, L. Mou, T. Yu, X. X. Zhu, and Z. J. Wang, "Dual adversarial network for unsupervised ground/satellite-to-aerial scene adaptation," in *ACM International Conference on Multimedia (ACMMM)*, 2020.
- [33] X. Wang, X. Xiong, and C. Ning, "Multi-label remote sensing scene classification using multi-bag integration," *IEEE Access*, vol. 7, pp. 120 399–120 410, 2019.
- [34] Q. Zhu, X. Sun, Y. Zhong, and L. Zhang, "High-resolution remote sensing image scene understanding: A review," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019.
- [35] Y. Long, G. Xia, S. Li, W. Yang, M. Yang, X. X. Zhu, L. Zhang, and D. Li, "DIRS: On creating benchmark datasets for remote sensing image interpretation," *arXiv:2006.12485*, 2020.
- [36] S. Zorzi and F. Fraundorfer, "Regularization of building boundaries in satellite images using adversarial and regularized losses," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019.
- [37] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2010.
- [38] J. Hu, T. Jiang, X. Tong, G. Xia, and L. Zhang, "A benchmark for scene classification of high spatial resolution remote sensing imagery," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2015.
- [39] Q. Zou, L. Ni, T. Zhang, and Q. Wang, "Deep learning based feature selection for remote sensing scene classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 11, pp. 2321–2325, 2015.
- [40] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1865–1883, 2017.
- [41] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [42] V. Risojević, S. Momić, and Z. Babić, "Gabor descriptors for aerial image classification," in *International Conference on Adaptive and Natural Computing Algorithms*, 2011.
- [43] M. Swain and D. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.
- [44] J. dos Santos, O. Penatti, and R. da Silva Torres, "Evaluating the potential of texture and color descriptors for remote sensing image retrieval and classification," in *International Conferences on Computer Vision Theory and Applications (VISAPP)*, 2010.

- [45] B. Manjunath and W. Ma, "Texture features for browsing and retrieval of image data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837–842, 1996.
- [46] V. Risojević and Z. Babić, "Aerial image classification using structural texture similarity," in *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 2011.
- [47] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*, 2010.
- [48] L. Zhao, P. Tang, and L. Huo, "Land-use scene classification using a concentric circle-structured multiscale bag-of-visual-words model," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 12, pp. 4620–4631, 2014.
- [49] M. Lienou, H. Maitre, and M. Datcu, "Semantic annotation of satellite images using latent dirichlet allocation," *IEEE Geoscience and Remote Sensing Letters*, vol. 7, no. 1, pp. 28–32, 2009.
- [50] Y. Zhong, Q. Zhu, and L. Zhang, "Scene classification based on the multifeature fusion probabilistic topic model for high spatial resolution remote sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 11, pp. 6207–6222, 2015.
- [51] F. Li, R. Feng, W. Han, and L. Wang, "High-resolution remote sensing image scene classification via key filter bank based on convolutional neural network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 11, pp. 8077–8092, 2020.
- [52] X. Lu, H. Sun, and X. Zheng, "A feature aggregation convolutional neural network for remote sensing scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 10, pp. 7894–7906, 2019.
- [53] R. Cao, L. Fang, T. Lu, and N. He, "Self-attention-based deep feature fusion for remote sensing scene classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 18, no. 1, pp. 43–47, 2020.
- [54] A. Zeggada, F. Melgani, and Y. Bazi, "A deep learning approach to UAV image multilabeling," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 5, pp. 694–698, 2017.
- [55] A. Zeggada, S. Benbraika, F. Melgani, and Z. Mokhtari, "Multilabel conditional random field classification for UAV images," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 3, pp. 399–403, 2018.
- [56] Y. Hua, L. Mou, and X. X. Zhu, "Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional LSTM network for multi-label aerial image classification," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 149, pp. 188–199, 2019.
- [57] G. Sumbul and B. Demir, "A deep multi-attention driven approach for multi-label remote sensing image classification," *IEEE Access*, vol. 8, pp. 95 934–95 946, 2020.
- [58] Y. Hua, L. Mou, and X. X. Zhu, "Relation network for multilabel aerial image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 7, pp. 4558–4572, 2020.
- [59] S. Koda, A. Zeggada, F. Melgani, and R. Nishii, "Spatial and structured SVM for multilabel image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 10, pp. 5948–5960, 2018.
- [60] B. Chaudhuri, B. Demir, S. Chaudhuri, and L. Bruzzone, "Multilabel remote sensing image retrieval using a semisupervised graph-theoretic method," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 2, pp. 1144–1158, 2017.
- [61] G. Xia, W. Yang, J. Delon, Y. Gousseau, H. Sun, and H. Maitre, "Structural high-resolution satellite image indexing," in *ISPRS TC VII Symposium*, 2010.
- [62] G. Sumbul, M. Charfuelan, B. Demir, and V. Markl, "BigEarthNet: A large-scale benchmark archive for remote sensing image understanding," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019.
- [63] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [64] J. Vert, K. Tsuda, and B. Schölkopf, "A primer on kernel methods," *Kernel Methods in Computational Biology*, vol. 47, pp. 35–70, 2004.
- [65] T. Ho, "Random decision forests," in *International Conference on Document Analysis and Recognition*, 1995.
- [66] T. Hastie, R. Tibshirani, and J. Friedman, "Boosting and additive trees," in *The Elements of Statistical Learning*, 2009.
- [67] J. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of Statistics*, pp. 1189–1232, 2001.
- [68] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv:1409.1556*, 2014.
- [69] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [70] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning (ICML)*, 2015.
- [71] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception architecture for computer vision," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [72] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, Inception-ResNet and the impact of residual connections on learning," in *AAAI Conference on Artificial Intelligence (AAAI)*, 2017.
- [73] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [74] F. Iandola, S. Han, M. Moskewicz, K. Ashraf, W. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and 0.5 MB model size," in *International Conference on Learning Representations (ICLR)*, 2017.
- [75] A. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv:1704.04861*, 2017.
- [76] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [77] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [78] N. Ma, X. Zhang, H. Zheng, and J. Sun, "ShuffleNetV2: Practical guidelines for efficient cnn architecture design," in *European Conference on Computer Vision (ECCV)*, 2018.
- [79] G. Huang, Z. Liu, L. V. D. Maaten, and K. Weinberger, "Densely connected convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [80] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [81] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. Zitnick, "Microsoft COCO: Common objects in context," in *European Conference on Computer Vision (ECCV)*, 2014.
- [82] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, and Q. Le, "MnasNet: Platform-aware neural architecture search for mobile," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [83] B. Zoph and Q. Le, "Neural architecture search with reinforcement learning," in *International Conference on Learning Representations (ICLR)*, 2017.
- [84] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and F. Li, "ImageNet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
- [85] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.
- [86] T. Ojala, M. Pietikainen, and T. Maenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.
- [87] Y. Lin, F. Lv, S. Zhu, M. Yang, T. Cour, K. Yu, L. Cao, and T. Huang, "Large-scale image classification: fast feature extraction and SVM training," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.
- [88] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *International Conference on Machine Learning (ICML)*, 2013.



Yuansheng Hua (S'18) received the bachelor's degree in remote sensing science and technology from the Wuhan University, Wuhan, China, in 2014, and double master's degrees in Earth Oriented Space Science and Technology (ESPACE) and Photogrammetry and remote sensing from the Technical University of Munich (TUM), Munich, Germany, and Wuhan University, Wuhan, China, in 2018 and 2019, respectively. He is currently pursuing the Ph.D. degree with the German Aerospace Center (DLR), Wessling, Germany and the Technical University of

Munich (TUM), Munich, Germany.

In 2019, he was a visiting researcher with the Wageningen University & Research, Wageningen, Netherlands. His research interests include remote sensing, computer vision, and deep learning, especially their applications in remote sensing.



Lichao Mou received the Bachelor's degree in automation from the Xi'an University of Posts and Telecommunications, Xi'an, China, in 2012, the Master's degree in signal and information processing from the University of Chinese Academy of Sciences (UCAS), China, in 2015, and the Dr.-Ing. degree from the Technical University of Munich (TUM), Munich, Germany, in 2020.

He is currently a Guest Professor at the Munich AI Future Lab AI4EO, TUM and the Head of Visual Learning and Reasoning team at the Department "EO Data Science", Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Wessling, Germany. Since 2019, he is a Research Scientist at DLR-IMF and an AI Consultant for the Helmholtz Artificial Intelligence Cooperation Unit (HAICU). In 2015 he spent six months at the Computer Vision Group at the University of Freiburg in Germany. In 2019 he was a Visiting Researcher with the Cambridge Image Analysis Group (CIA), University of Cambridge, UK.

He was the recipient of the first place in the 2016 IEEE GRSS Data Fusion Contest and finalists for the Best Student Paper Award at the 2017 Joint Urban Remote Sensing Event and 2019 Joint Urban Remote Sensing Event.



Pu Jin (S'21) received the bachelor's degree in electronic information science and technology from the Wuhan University, Wuhan, China, in 2017, and double master's degrees in Earth Oriented Space Science and Technology (ESPACE) and Photogrammetry and remote sensing from the Technical University of Munich (TUM), Munich, Germany, and Wuhan University, Wuhan, China, in 2020 and 2021, respectively. He is currently pursuing the Ph.D. degree with the German Aerospace Center (DLR), Wessling, Germany and the Technical University of

Munich (TUM), Munich, Germany. His research interests include remote sensing, computer vision, and deep learning, especially their applications in remote sensing.



Xiao Xiang Zhu (S'10–M'12–SM'14–F'21) received the Master (M.Sc.) degree, her doctor of engineering (Dr.-Ing.) degree and her "Habilitation" in the field of signal processing from Technical University of Munich (TUM), Munich, Germany, in 2008, 2011 and 2013, respectively.

She is currently the Professor for Data Science in Earth Observation (former: Signal Processing in Earth Observation) at Technical University of Munich (TUM) and the Head of the Department "EO Data Science" at the Remote Sensing Technology Institute, German Aerospace Center (DLR). Since 2019, Zhu is a co-coordinator of the Munich Data Science Research School (www.muds.de). Since 2019 She also heads the Helmholtz Artificial Intelligence – Research Field "Aeronautics, Space and Transport". Since May 2020, she is the director of the international future AI lab "AI4EO – Artificial Intelligence for Earth Observation: Reasoning, Uncertainties, Ethics and Beyond", Munich, Germany. Since October 2020, she also serves as a co-director of the Munich Data Science Institute (MDSI), TUM. Prof. Zhu was a guest scientist or visiting professor at the Italian National Research Council (CNR-IREA), Naples, Italy, Fudan University, Shanghai, China, the University of Tokyo, Tokyo, Japan and University of California, Los Angeles, United States in 2009, 2014, 2015 and 2016, respectively. She is currently a visiting AI professor at ESA's Phi-lab. Her main research interests are remote sensing and Earth observation, signal processing, machine learning and data science, with a special application focus on global urban mapping.

Dr. Zhu is a member of young academy (Junge Akademie/Junges Kolleg) at the Berlin-Brandenburg Academy of Sciences and Humanities and the German National Academy of Sciences Leopoldina and the Bavarian Academy of Sciences and Humanities. She serves in the scientific advisory board in several research organizations, among others the German Research Center for Geosciences (GFZ) and Potsdam Institute for Climate Impact Research (PIK). She is an associate Editor of IEEE Transactions on Geoscience and Remote Sensing and serves as the area editor responsible for special issues of IEEE Signal Processing Magazine. She is a Fellow of IEEE.

E Yuansheng Hua, Diego Marcos, Lichao Mou, Xiao Xiang Zhu, and Devis Tuia, “Semantic segmentation of remote sensing images with sparse annotations,” *IEEE Geoscience and Remote Sensing Letters*, in press, 2021.

<https://doi.org/10.1109/LGRS.2021.3051053>

Semantic Segmentation of Remote Sensing Images with Sparse Annotations

Yuansheng Hua, Diego Marcos, Lichao Mou, Xiao Xiang Zhu, *Fellow, IEEE*, Devis Tuia, *Senior Member, IEEE*

Abstract—This is the preprint version. To read the final version, please go to [IEEE Geoscience and Remote Sensing Letters](#). Training Convolutional Neural Networks (CNNs) for very high resolution images requires a large quantity of high-quality pixel-level annotations, which is extremely labor- and time-consuming to produce. Moreover, professional photo interpreters might have to be involved for guaranteeing the correctness of annotations. To alleviate such a burden, we propose a framework for semantic segmentation of aerial images based on incomplete annotations, where annotators are asked to label a few pixels with easy-to-draw scribbles. To exploit these sparse scribbled annotations, we propose the FEature and Spatial relaTional regulARization (FESTA) method to complement the supervised task with an unsupervised learning signal that accounts for neighbourhood structures both in spatial and feature terms. For the evaluation of our framework, we perform experiments on two remote sensing image segmentation datasets involving aerial and satellite imagery, respectively. Experimental results demonstrate that the exploitation of sparse annotations can significantly reduce labeling costs while the proposed method can help improve the performance on semantic segmentation when training on such annotations. The sparse labels and codes are publicly available for reproducibility purposes¹.

Index Terms—Semantic segmentation, aerial image, sparse scribbled annotation, convolutional neural networks, semi-supervised learning.

I. INTRODUCTION

Semantic segmentation of remote sensing imagery aims at identifying the land-cover or land-use category of each pixel in an image. As one of the fundamental visual tasks, semantic segmentation has been attracting wide attention in the remote sensing community and proven to be beneficial to a variety of applications, such as land cover mapping, traffic monitoring and urban management. Recently, many studies [1] resort to learning deep Convolutional Neural Networks (CNNs) with full supervision for semantic segmentation and have obtained enormous achievements. However, training a fully supervised segmentation CNN requires a huge volume of dense pixel-level ground truths, which are labor- and time-consuming to generate. Moreover, expert annotators might be needed for correctly identifying pixels located at object boundaries and ambiguous regions (e.g., shadows in Fig. 1).

YH, LM and XXZ are with Data Science in Earth Observation, Technical University of Munich, Germany, and Remote Sensing Technology Institute, German Aerospace Center, Germany. (e-mails: yuansheng.hua@dlr.de; lichao.mou@dlr.de; xiaoxiang.zhu@dlr.de) DM is/DT was with the Laboratory of GeoInformation Science and Remote Sensing, Wageningen University, the Netherlands. (e-mail: diego.marcos@wur.nl). He is now with the Ecole Polytechnique Fédérale de Lausanne, Sion, Switzerland. (e-mail: devis.tuia@epfl.ch)(Correspondences: Xiao Xiang Zhu, Devis Tuia.)

The work is supported by the German Federal Ministry of Education and Research – AI future lab “AI4EO” (Grant number: 01DD20001).

¹<https://github.com/Hua-YS/Semantic-Segmentation-with-Sparse-Labels>

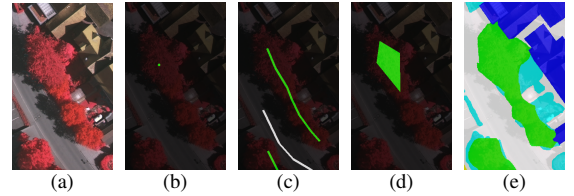


Fig. 1. Comparisons of different levels of scribbled annotations. Trees (marked as green) are taken as an example here. Images from left to right are (a) an aerial image, (b) point-, (c) line- and (d) polygon-level scribbled annotations, and (e) dense pixel-wise labels.

To alleviate the requirement of dense pixel-wise annotations, semi-supervised learning approaches are proposed to make use of additional information, such as spatial relations (e.g. neighboring pixels are likely to belong to the same class) or feature-level relations (e.g. pixels with similar CNN feature representations are likely to belong to the same class), for semantic segmentation. These methods aim to utilize low-cost annotations, such as points [2], scribbles [3], [4] or image-level labels [5], [6]. As the first attempt, Bearman *et al.* [2] proposed to learn semantic segmentation models with point-level supervision, where only one point is labeled for each instance. In scribble-supervised algorithms, annotations are provided in the form of hand-drawn scribbles. Wu *et al.* [3] propose to learn aerial building footprint segmentation models from scribbles. Maggilo *et al.* [4] argue that a network directly trained on scribbled ground truths fails to accurately predict object boundaries and propose to employ a fully connected Conditional Random Field (CRF) to refine the shapes of objects. Compared to fully annotated ground truths, scribbled annotations (cf., Fig. 1(c)) are easier to generate in a user-friendly way. In comparison with point-level annotations (e.g., Fig. 1(b)), scribbles can provide stronger supervisory signals. However, point- and scribble-supervised segmentation methods remain under-explored in the remote sensing community. To this end, we propose a simple yet effective framework for semantic segmentation of remote sensing imagery with low-cost annotations. In this framework, we manually create point- or scribble-level annotations and train networks on them. Besides, we also evaluate polygon-level annotations (see Fig. 1(d)), which can be easily yielded and cover more pixels than the other types of annotations. Since these annotations are sparsely distributed across the images, we call them sparse annotations in the following sections. In order to better exploit sparse annotations, we propose a semi-supervised learning method which encodes and regularizes the feature and spatial relations. To demonstrate the effectiveness of our learning framework, extensive experiments are conducted on two VHR datasets, the Vaihingen and Zurich Summer.

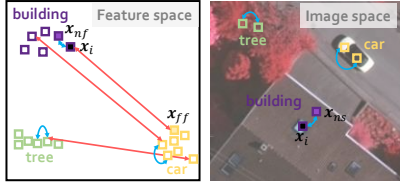


Fig. 2. Illustration of the proposed FESTA. A Sample x_i belonging to *building* (filled with black) is taken as an example.

II. METHODOLOGY

A. Supervision with Sparse Annotations

In contrast to conventional dense annotations, sparse annotations have two characteristics: 1) a very small proportion pixels are assigned semantic classes, and 2) objects do not need to be entirely annotated (see (b), (c), (d) in Fig. 1). This greatly reduces the effort required from the annotators, as complex boundaries and ambiguous pixels can be avoided.

Here we consider three levels of sparse annotations: point-, scribble-, and polygon-level. Specifically, point-level annotations indicate that, for an annotator interaction, only one single pixel is labeled. Scribble-level annotations, also called line-level annotations, are yielded by drawing a scribble line within an object and assigning all pixels along this line the same class label. Similarly, polygon-level annotations can be generated by drawing a polygon within an object and classifying pixels located in the polygon into the same semantic class. Examples of these three levels of annotations are shown in Fig. 1.

B. Feature and Spatial Relational Regularization

When using sparse annotations, the vast majority of pixels in the training images are left unlabelled. In order to exploit both labeled and unlabeled pixels, we develop a semi-supervised methodology, named FEature and Spatial relational regularization (FESTA), to enable a semantic segmentation CNN to learn discriminative features, while leveraging the unlabelled image pixels. An assumption shared by many unsupervised learning algorithms [7] is that nearby entities often belong to the same class. Based on this assumption, a recent work [8] achieves success in representation learning by encoding neighborhood-relations in the feature space. Inspired by this work, we propose to encode and regularize relations between pixels in both feature and spatial domain, as shown in Fig. 2, so that the learned features become more useful for semantic segmentation.

Specifically, given a sample x_i (i.e., a CNN feature vector extracted from location i in an image), we first encode its relations to all other samples by measuring the distance in space and feature similarity with respect to all other features in the image. The sample with the smallest similarity is considered as the far-away sample in the feature space, x_{iff} , while that with the highest similarity is defined as the neighboring sample in feature space, x_{inf} . According to the aforementioned proximity assumption, it is highly probable that x_i and x_{inf} belong to the same class, and thus, the distance between them should be as small as possible. In order to prevent a trivial solution in which all features collapse to the same point, x_i and x_{iff} are encouraged to further increase

their dissimilarity. We apply a similar reasoning in the spatial domain, since images are smooth in spatial terms. Thus, we take the 8 spatial neighbors of x_i into consideration and chose the one most similar in feature space as the spatial neighbor, x_{ins} . This operation is intended to prevent pairing x_i with a spatial neighbor that belongs to the object boundary.

These priors can be incorporated into the learning objectives by using the following loss function:

$$\mathcal{L}_{FESTA} = \alpha \sum_{i=1}^N \mathcal{D}(x_i, x_{inf}) + \beta \sum_{i=1}^N \mathcal{D}(x_i, x_{ins}) + \gamma \sum_{i=1}^N \mathcal{S}(x_i, x_{iff}), \quad (1)$$

where \mathcal{D} denotes the euclidean distance and \mathcal{S} represents cosine similarity. α , β , and γ are trade-off parameters representing the significances of the respective terms, and N represents the number of pixels in a given image. By minimizing \mathcal{L}_{FESTA} , x_{inf} and x_{ins} are forced to move closer to x_i , while x_{iff} is pushed far from x_i . In order to jointly exploit the sparse scribbled annotations and FESTA for the network training, the final loss is defined as:

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda \mathcal{L}_{FESTA}, \quad (2)$$

where \mathcal{L}_{ce} indicates the categorical cross-entropy loss calculated from pixels with annotations.

C. CRF for Boundary Refinement

To further refine the predictions of networks trained on scribbled annotations, we integrate a fully connected CRF [9] into our system, and the energy function of CRF model is

$$E = \sum_i \theta_u(x_i) + \sum_{ij} \theta_p(x_i, x_j), \quad (3)$$

where $\theta_u(x_i)$ is the unary potential and calculated as $\theta_u(x_i) = -\log P(x_i)$. Here i ranges from 0 to the number of pixels in the image, and $P(x_i)$ is the label probability of pixel i . $\theta_p(x_i, x_j)$ is utilized to measure pairwise potentials between pixel i and j . We tested with two Gaussian kernels,

$$k_1 = \exp\left(-\frac{\|p_i - p_j\|^2}{2\theta_1^2} - \frac{\|I_i - I_j\|^2}{2\theta_2^2}\right), \quad (4)$$

$$k_2 = \exp\left(-\frac{\|p_i - p_j\|^2}{2\theta_3^2}\right),$$

where p_i and I_i indicate the position and color intensity of pixel i . θ_1 , θ_2 , and θ_3 are hyperparameters that control the kernel "scale". In Eq. 4, k_1 is known as *appearance kernel* and tends to classify neighboring pixels with similar appearances [10], i.e., color intensities, into the same classes, while k_2 , so-called *smoothness kernel*, penalizes pixels nearby but assigned diverse labels. This step is expected to make the class map smoother within homogeneous areas.

III. EXPERIMENTAL RESULTS

A. Dataset Description

The Vaihingen dataset² is a benchmark dataset for semantic segmentation provided by the International Society for Photogrammetry and Remote Sensing (ISPRS). 33 aerial images

²<http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-vaihingen.html>

TABLE I

THE TOTAL NUMBERS OF PIXELS LABELED WITH SPARSE POINT-, LINE-, AND POLYGON-LEVEL ANNOTATIONS (MIDDLE THREE COLUMNS) AND DENSE ANNOTATIONS (RIGHT COLUMN) IN THE VAIHINGEN AND ZURICH SUMMER DATASETS.

Dataset Name	Point	Line	Polygon	Dense*
Vaihingen	18,787	480,593	4,591,409	54,373,518
Zurich Summer	29,508	330,767	1,445,270	12,266,287

*Background/Clutter is not considered.

with a spatial resolution of 9 cm were collected over the city of Vaihingen, and each image covers an average area of 1.38 km². For each aerial image, three bands are available, near infrared (NIR), red (R), and green (G). Besides, coregistered digital surface models (DSMs) are provided for all images. 16 images are fully annotated. In total, six land-cover classes are considered: impervious surface, building, low vegetation, tree, car, and clutter/background. In this paper, we follow the train-test split scheme in most existing works [11], [12] and select five images (image IDs: 11, 15, 28, 30, 34) as the test set. The remaining ones are utilized to train our models.

The Zurich Summer dataset [13] is composed of 20 images, which are taken over the city of Zurich in August 2002 by the QuickBird satellite. The spatial resolution is 0.62 m, and the average size of images is 1,000 × 1,150 pixels. The images consist of four channels: near infrared (NIR), red (R), green (G), and blue (B). Following previous works [14], [15], we only utilize NIR, R, and G in our experiments and train our model on 15 images; the others (image IDs: 16, 17, 18, 19, 20) are utilized to test. In total, there are 8 urban classes, including road, building, tree, grass, bare soil, water, railway, and swimming pool. Uncategorized pixels are labeled as background.

It is noteworthy that although full pixel-wise annotations are provided for all images in the Vaihingen and Zurich Summer dataset, we only use them in the test phase to calculate evaluation metrics. The training of all models is done with scribbled annotations described below.

B. Scribbled Annotation Generation

To annotate large-scale images, we employ an online labeling platform, LabelMe³, and ask annotators to draw by following these rules: 1) for each class, annotations are supposed to cover diverse appearances (see region a, b, and c in Figure 3, where cars of different colors are annotated) and be located in different positions of the image separately. 2) polygon- and line-level annotations are not required to delineate object boundaries precisely, see the annotations of trees in Fig. 1(c) and 1(d). In order to make the time spent on each level of scribbled annotations more equivalent, we ask 4 annotators (including 2 non-experts) to label 7, 5, and 3 objects per class for point-, line- and polygon-level annotations in each aerial image. As a consequence, sparse but accurate annotations can be provided rapidly without effort. Since a point- or line-level annotation is often located in the centre area of an object and distant from its boundary, we perform morphological dilation on all point- and line-level annotations

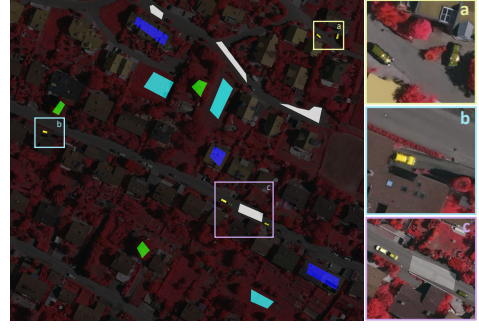


Fig. 3. Example polygon-level annotations of an image (ID: 13) on the Vaihingen dataset. Annotations of cars are zoomed in to illustrate that annotations should include variant visual appearances for one class. Legend—white: impervious surfaces, blue: buildings, cyan: low vegetation, green: trees, yellow: cars.

with a disk of radius 3. Afterwards, pixels involved in dilated annotations are assigned the same class labels as their central points or lines. For polygon-level annotations, pixels within each polygon are assigned the corresponding classes.

Table I shows the average amounts of pixels with sparse and dense annotations in both datasets. It can be seen that sparse annotations are several orders of magnitude fewer than dense annotations. As to the labeling time, it took on average 133, 126, and 161 seconds per image to produce point-, line- and polygon-level annotations, respectively, for the Vaihingen dataset, and 177, 162, and 238 seconds per image for the Zurich Summer dataset. In Section III-D, we demonstrate the proposed method allows to improve the semantic segmentation results using these sparse annotations. In Section III-D, we discuss the differences observed among the tested annotation types.

C. Training Details

We segment the images with a standard FCN (i.e., FCN-16s [17]) and initialize convolutional layers with Glorot uniform [18] initializers. Specifically, VGG-16 is taken as the backbone, and outputs of the last two convolutional blocks are upsampled to the original resolution and fused with an element-wise addition. The fused feature maps are finally fed into a convolutional layer, where the number of filters is equivalent to the number of classes. In the training phase, all weights are trainable and updated with Nesterov Adam [19], using $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e-08$ as recommended. We initialize the learning rate as $2e-04$ and let it decay by a factor of 10 when validation loss is saturated. To train the network, we define the loss as Eq. 2, and λ is set experimentally to 0.1 and 0.01 for the Vaihingen and Zurich Summer datasets, respectively. Tradeoff parameters, α , β , and γ , are set as 0.5, 1.5, and 1, to ensure that 1) the regularizers governing feature and spatial relations are balanced, and 2) neighboring pixels in the image space receive more attention. The network, as well as FESTA, is implemented on TensorFlow and trained on one NVIDIA Tesla P100 16GB GPU for 100k iterations. The size of mini-batch is set as 5 during the training procedure. In the training phase, we use a sliding window to crop training images into 256×256 patches, and its stride is set to 64 pixels. Besides, no class-dependent configurations are considered. In

³<http://labelme.csail.mit.edu/Release3.0/>

TABLE II

NUMERICAL RESULTS ON THE VAIHINGEN DATASET (%): WE SHOW THE PER-CLASS F_1 SCORE, MEAN F_1 SCORE, AND OVERALL ACCURACY ON THE TEST SET. MEAN AND STANDARD DEVIATION OF EACH METRIC ARE CALCULATED FROM RESULTS ON SPARSE ANNOTATIONS PRODUCED BY 4 ANNOTATORS. RESULTS ON DENSE ANNOTATIONS ARE PROVIDED AS REFERENCE.

Scribble	Model	Imp. surf.	Build.	Low veg.	Tree	Car	mean F_1	OA
Point	FCN-WL [16]	69.81 ± 1.52	75.02 ± 2.32	60.25 ± 3.40	76.17 ± 1.42	12.29 ± 3.60	58.71 ± 0.33	67.11 ± 0.97
	FCN+dCRF [4]	75.37 ± 0.93	81.37 ± 3.10	61.93 ± 5.54	78.50 ± 1.69	17.51 ± 6.70	62.94 ± 0.44	72.53 ± 0.42
	FCN-FESTA	74.65 ± 2.73	78.64 ± 4.74	60.24 ± 3.33	76.15 ± 2.07	23.65 ± 4.24	62.66 ± 2.54	71.43 ± 2.93
	FCN-FESTA+dCRF	77.62 ± 1.93	80.08 ± 5.27	60.78 ± 4.00	76.70 ± 2.00	31.40 ± 5.24	65.32 ± 2.56	73.65 ± 2.52
Line	FCN-WL [16]	78.44 ± 3.24	83.45 ± 1.58	64.02 ± 2.34	79.32 ± 0.54	29.01 ± 2.96	66.85 ± 1.81	76.12 ± 1.52
	FCN+dCRF [4]	81.32 ± 2.45	84.88 ± 1.88	63.71 ± 3.92	79.88 ± 1.33	38.95 ± 4.50	69.75 ± 2.23	78.03 ± 1.82
	FCN-FESTA	78.12 ± 3.92	83.76 ± 2.00	65.78 ± 1.88	80.49 ± 0.93	38.24 ± 10.31	69.28 ± 3.66	77.24 ± 2.27
	FCN-FESTA+dCRF	80.06 ± 3.32	84.47 ± 2.23	64.35 ± 2.38	80.32 ± 0.92	43.72 ± 9.62	70.58 ± 3.42	77.99 ± 2.14
Polygon	FCN-WL [16]	76.71 ± 3.63	80.03 ± 1.42	59.40 ± 6.09	78.50 ± 2.86	26.28 ± 11.06	64.19 ± 4.40	74.18 ± 2.97
	FCN+dCRF [4]	78.37 ± 3.08	80.85 ± 1.13	57.92 ± 7.67	78.67 ± 2.87	29.13 ± 8.15	64.99 ± 3.99	75.15 ± 2.94
	FCN-FESTA	78.98 ± 3.82	83.10 ± 2.62	62.59 ± 4.89	79.91 ± 3.31	33.04 ± 7.71	67.52 ± 4.07	76.65 ± 3.39
	FCN-FESTA+dCRF	80.62 ± 3.22	83.62 ± 2.29	60.79 ± 5.04	79.81 ± 2.52	40.27 ± 8.30	69.02 ± 4.01	77.32 ± 2.92
Dense	FCN [17]	88.67	92.83	76.32	74.21	86.67	83.74	86.51

TABLE III

NUMERICAL RESULTS ON THE ZURICH SUMMER DATASET (%): WE SHOW THE PER-CLASS F_1 SCORE, MEAN F_1 SCORE, AND OVERALL ACCURACY ON THE TEST SET. MEAN AND STANDARD DEVIATION OF EACH METRIC ARE CALCULATED FROM RESULTS ON SPARSE ANNOTATIONS PRODUCED BY 4 ANNOTATORS. RESULTS ON DENSE ANNOTATIONS ARE PROVIDED AS REFERENCE.

Scribble	Model	Road	Build.	Tree	Grass	Soil	Water	Rail.	Pool	mean F_1	OA
Point	FCN-WL [16]	69.74±3.98	78.94±3.01	82.33±2.55	82.20±2.40	53.37±7.03	87.87±1.40	0.81±1.42	48.89±9.42	63.02±2.14	77.38±2.73
	FCN+dCRF [4]	72.13 ±4.99	80.71 ±1.84	82.87±2.08	83.55±2.07	63.92±8.90	92.71±1.26	2.09 ±4.17	59.96±14.60	67.24±1.93	80.03 ±2.26
	FCN-FESTA	70.64±3.44	77.34±4.13	82.91 ±2.48	83.73±2.34	56.67±5.64	89.67±2.25	0.94±1.89	73.62±4.06	66.94±2.56	78.17±3.00
	FCN-FESTA+dCRF	71.23±2.61	77.71±3.17	82.81±1.99	84.18 ±1.96	66.34 ±3.69	93.40 ±1.81	0.00±0.00	77.38 ±8.87	69.05 ±1.15	79.11±2.14
Line	FCN-WL [16]	73.00±4.60	81.17 ±3.77	82.82 ±2.78	81.88±1.41	67.02±8.77	90.98±1.79	1.19±1.60	58.77±7.82	67.10±2.02	79.75 ±2.25
	FCN+dCRF [4]	71.71±4.83	79.22±4.01	81.22±3.06	80.43±2.10	71.72 ±9.20	84.65±14.90	2.35 ±4.71	67.58±17.39	68.39±3.10	78.84±2.15
	FCN-FESTA	73.34 ±3.88	79.08±3.60	82.71±2.10	84.27 ±1.41	60.67±13.36	92.37±1.44	1.02±0.83	74.27±8.24	68.47±2.45	79.52±2.86
	FCN-FESTA+dCRF	71.74±2.78	75.81±4.18	81.20±1.60	83.44±1.51	66.49±15.57	94.68 ±0.52	0.00±0.00	82.06 ±6.80	69.43 ±2.57	78.51±2.21
Polygon	FCN-WL [16]	64.18±6.14	72.17±6.01	79.64±4.25	77.10±3.92	49.17±16.96	89.26±3.52	1.31±1.09	76.90±6.33	63.72±4.35	73.09±4.49
	FCN+dCRF [4]	62.63±5.77	70.35±4.88	78.30±3.53	75.94±4.42	52.11±14.06	91.03±4.39	0.84±1.69	85.13 ±2.72	64.54±4.08	72.37±3.89
	FCN-FESTA	66.53 ±5.07	74.06 ±3.06	80.05 ±3.66	79.42 ±3.56	57.83±11.38	90.80±2.42	5.87±4.86	65.68±16.06	65.03±1.98	75.00 ±3.17
	FCN-FESTA+dCRF	65.10±4.42	71.96±2.76	79.44±3.26	78.87±4.58	61.86 ±9.72	92.50 ±2.96	6.37 ±6.63	77.21±6.63	66.66 ±2.41	74.41±2.86
Dense	FCN [17]	88.34	93.27	92.40	89.48	67.96	96.87	2.98	88.10	77.42	90.51

the test phase, we employ dense CRF to refine predictions before calculating metrics. We tuned the parameters of dense CRF (θ_1 , θ_2 , and θ_3 in Eq. 4) on validation images, and find that satisfactory results can be achieved for both FCN and FCN-FESTA when setting them to 30, 10, and 10, respectively. In the case of large homogeneous areas of an image belonging to the same class, α should be set to a small value, which encourages the network to focus more on geographically nearby samples. Besides, a large batch size and sliding window can also help alleviate the influence of such a scenario.

D. Comparing with Existing Methods

We compare a Fully Convolutional Network [17] (FCN) learned using the proposed FESTA (FCN-FESTA) against an FCN learned with weighted loss function (FCN-WL) [16] on sparse annotations. We also report segmentation results of the baseline FCN trained on dense labels. In addition, we study the influence of the fully connected CRF by comparing FCN-FESTA+dCRF and FCN+dCRF [4]. Each model is trained and validated on sparse annotations independently. Per-class F_1 scores, mean F_1 scores, and overall accuracy (OA) are calculated on test images with dense annotations. Considering that each model is learned on labels from four annotators, respectively, we average metrics obtained by each annotator and report them in the form of mean ± standard deviation.

Table II exhibits numerical results on the Vaihingen dataset. FCN-FESTA+dCRF achieves the highest mean F_1 scores in training with all kinds of scribbled annotations, which demonstrates its effectiveness. To be more specific, with point-

and polygon-level supervision, FCN-FESTA improves the mean F_1 score by 3.95% and 3.33% compared to FCN-WL, respectively. By refining predictions with dense CRF, FCN-FESTA+dCRF achieves improvements of 2.38% and 4.03% in comparison with FCN+dCRF. It is interesting to observe that line-level scribbles improve the segmentation performance the most, and FCN-FESTA+dCRF learned with such annotations obtains the highest mean F_1 score, 70.58%. Moreover, we note that FESTA can enhance the network capability of recognizing small objects, i.e., *car*, in high resolution aerial images. Example segmentation results of networks trained on line annotations are visualized in Fig. 4.

Numerical results on the Zurich Summer dataset are shown in Table III. As can be seen, FESTA contributes to increments of 3.92%, 1.37% and 1.31% in the mean F_1 score when training with point-, line- and polygon-level annotations. By utilizing line annotations and dense CRF, FCN-FESTA+dCRF obtains the highest mean F_1 score, 69.43%. Besides, we note that the exploitation of dense CRF plays a significant role in improving results of networks trained on point-level scribbles. Example visual results of networks trained on line annotations are shown in Fig. 5. In our experiments, we also train networks with multi-class dice loss and find that results are comparative to those learned with crossentropy loss.

E. Discussion on annotation type

To further study the influence of annotations, we also train baseline FCNs on dense annotations and report numerical results in Tables II and III. As shown in Tables II and III,

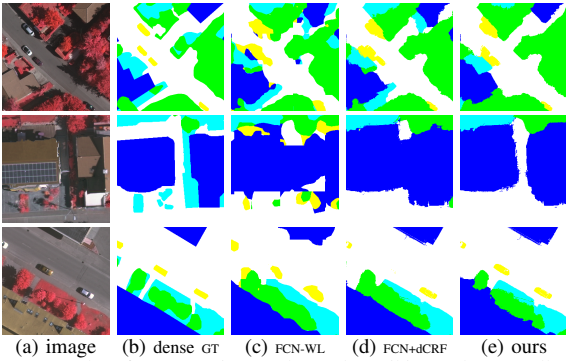


Fig. 4. Examples of segmentation results on the Vaihingen dataset. All models are trained on line annotations. The legend is the same as that in Fig. 3.

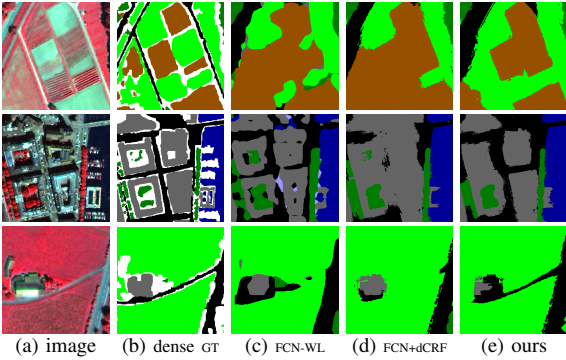


Fig. 5. Examples of segmentation results on the Zurich Summer dataset. All models are trained on line annotations. Legend—black: road, brown: soil, green: grass, dark green: tree, gray: building, and white: background.

line-level annotations lead to the best performance on both datasets, even though the number of labeled pixels is an order of magnitude smaller than polygon annotations (see Table I). Although it was expected that line annotations would outperform point annotations, due to their ability to capture within-object variations, we were surprised to see that they also outperformed polygon annotations. We suspect that this is linked to the fact that the number of pixels per object grows quadratically for polygons and linearly for lines. This would lead to a more balanced weighing of differently sized objects in the case of line annotations and an under-weighing of smaller objects in the case of polygon annotations, which could harm the model's performance. Another reason could be that, since drawing a line is faster than drawing a polygon, annotators for the line features provided more scribbles in the same time budget.

In spite of the mean F_1 performance boost provided by FESTA, there is still a large gap with respect to the FCN model trained with dense ground truths, of 13% in Vaihingen and 8% in Zurich. This gap is, however, not evenly distributed across the classes. The gap is smaller or non-existent in classes such as water, tree, grass or soil, which are often homogeneous in terms of materials. On the contrary, it is larger for classes with more diverse materials (and therefore observed spectral values), such as building and car (in the Vaihingen dataset). It is noteworthy to mention that the class railway, in the Zurich dataset, is systematically missed in all cases, including the densely supervised FCN.

IV. CONCLUSION

In this paper, we propose a simple yet efficient framework for semantic aerial image segmentation using sparse annotations and a semi-supervised learning objective. In order to validate the effectiveness of our approach, we conduct experiments on the Vaihingen and Zurich Summer datasets. Numerical and visual results suggest that the proposed method contributes to the improvement of semantic segmentation results using several kinds of sparse annotations. Although models learned on sparse annotations achieve relatively lower accuracies than those using dense annotations, we show that using a semi-supervised deep learning approach can help closing this performance gap while leveraging sparse annotations that can significantly reduce the costs of label generation. As future work, the proposed framework can be further improved by introducing graph-based models and prior knowledge learned from label semantics.

ACKNOWLEDGMENT

The authors would like to thank Yingya Xu, Li Hua, and Yanping Tang for contributing to this work with annotation.

REFERENCES

- [1] X. X. Zhu, D. Tuia, L. Mou, G. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," *GRSM*, vol. 5, no. 4, pp. 8–36, 2017.
- [2] A. Bearman, O. Russakovsky, V. Ferrari, and F. Li, "What's the point: Semantic segmentation with point supervision," in *ECCV*, 2016.
- [3] W. Wu, H. Qi, Z. Rong, L. Liu, and H. Su, "Scribble-supervised segmentation of aerial building footprints using adversarial learning," *IEEE Access*, vol. 6, pp. 58 898–58 911, 2018.
- [4] L. Maggiori, D. Marcos, G. Moser, and D. Tuia, "Improving maps from CNNs trained with sparse, scribbled ground truths using fully connected CRFs," in *IGARSS*, 2018.
- [5] A. Nivaggioli and H. Randrianarivo, "Weakly supervised semantic segmentation of satellite images," in *JURSE*, 2019.
- [6] R. Zhu, L. Yan, N. Mo, Y. Liu, "Semi-supervised center-based discriminative adversarial learning for cross-domain scene-level land-cover classification of aerial images," *ISPRS P & RS*, vol. 155, pp. 72–89, 2019.
- [7] T. Cover and P. Hart, "Nearest neighbor pattern classification," *TIT*, vol. 13, no. 1, pp. 21–27, 1967.
- [8] M. Sabokrou, M. Khalooei, and E. Adeli, "Self-supervised representation learning via neighborhood-relational encoding," in *ICCV*, 2019.
- [9] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected CRFs with gaussian edge potentials," in *NeurIPS*, 2011.
- [10] K. Schindler, "An overview and comparison of smooth labeling methods for land-cover classification," *TGRS*, vol. 50, no. 11, pp. 4534–4545, 2012.
- [11] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "High-resolution aerial image labeling with convolutional neural networks," *TGRS*, vol. 55, no. 12, pp. 7092–7103, 2017.
- [12] J. Sherrah, "Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery," *arXiv:1606.02585*, 2016.
- [13] M. Volpi and V. Ferrari, "Semantic segmentation of urban scenes by learning local class interactions," in *CVPRw*, 2015.
- [14] D. Tuia, M. Volpi, and G. Moser, "Decision fusion with multiple spatial supports by conditional random fields," *TGRS*, vol. 56, no. 6, pp. 3277–3289, 2018.
- [15] C. Wendt, D. Marcos, and D. Tuia, "Novelty detection in very high resolution urban scenes with density forests," in *JURSE*, 2019.
- [16] Ö. Çiçek, A. Abdulkadir, S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: Learning dense volumetric segmentation from sparse annotation," in *MICCAI*, 2016.
- [17] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015.
- [18] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *AISTATS*, 2010.
- [19] T. Dozat, "Incorporating Nesterov momentum into Adam," in *International Conference on Learning Representations Workshop*, May 2016.