Peaks Fusion assisted Early-stopping Strategy for Overhead Imagery Segmentation with Noisy Labels

Chenying Liu[1,2], Conrad M Albrecht[2], Yi Wang[1,2], Xiao Xiang Zhu[1]
[1]*Chair of Data Science in Earth Observation, Technical University of Munich (TUM), Germany*
[2]*Remote Sensing Technology Institute, German Aerospace Center (DLR), Germany*

***Abstract*—Automatic label generation systems, which are capable to generate huge amounts of labels with limited human efforts, enjoy lots of potential in the deep learning era. These easy-to-come-by labels inevitably bear label noises due to a lack of human supervision and can bias model training to some inferior solutions. However, models can still learn some plausible features, before they start to overfit on noisy patterns. Inspired by this phenomenon, we propose a new Peaks fusion assisted EArly-Stopping (PEAS) approach for imagery segmentation with noisy labels, which is mainly composed of two parts. First, a fitting based early-stopping criterion is used to detect the turning phase from which models are about to mimic noise details. After that, a peaks fusion strategy is applied to select reliable models in the detection zone to generate final fusion results. Here, validation accuracies are utilized as indicators in model selection. The proposed method was evaluated on New York City dataset whose labels were automatically collected by a rule-based label generation system, thus noisy to some extent due to a lack of human supervision. The experimental results showed that the proposed PEAS method can achieve both promising statistical and visual results when trained with noisy labels.**

***Index Terms*—deep learning, semantic segmentation, noisy labels, early stopping**

I. Introduction

Land cover information is critical for various real applications such as urban planning [1], natural disaster monitoring [7], and so on [2]. Specifically, nearly real-time generation of such information would further benefit many of these applications, thereby essential for digital twins construction. Traditional field survey based land cover mapping methods are generally laborious, time-consuming, and expensive, hard to meet the demand. Fortunately, due to the rapid development of remote sensing and machine learning (deep learning in particular) techniques, we are able to access as well as cope with massive amounts of remote sensing data, making it possible to yield land cover maps from a large scale in a real-time manner [6]. However, obtaining a sufficient number of labeled data for model training is challenging, since they mainly rely on costly human annotations like visual interpretation and in-situ field surveys.

To address this problem, researchers seek to find solutions by developing automatic labeling tools. For example, Albrecht *et al.* [8] designed an automatic label generation system named *AutoGeoLabel*. With some high-quality data e.g. LiDAR (Light Detection And Ranging) data in hand, this system can easily distinguish different kinds of ground objects based on simple rules following daily common sense. One of the biggest advantages in this process is to spare human efforts, realizing automatic and near real-time label collection. In spite of the convenience and speed, these rule-based labels inevitably bear some noises, threatening to bias the model training [9].

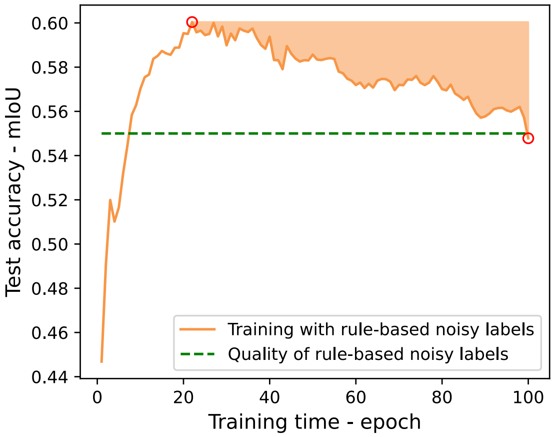
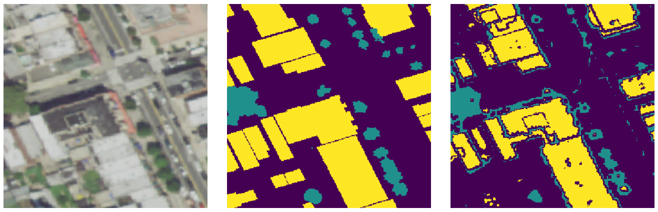


Fig. 1: Test accuracies (mIoUs) versus training time (epochs) obtained by training with rule-based noisy labels.

As mentioned in [4], deep classification networks are capable of modeling noisy labels, yet they tend to first learn simple/common features, and then start to mimic noisy patterns as training proceeds. In this case, the model performance gets improved merely at the initial stage, and gradually decreases due to an overfitting to noisy labels. This phenomenon termed as *memorization effect* has also been reported on segmentation tasks [10]. Here we plot test mIoUs (mean Intersection over Union) by the model learned from rule-based noisy labels as a function of training time in Fig. 1. As illustrated, models themselves have the competence of ”filtering” some label noises, likely to achieve better segmentation results than the quality of original noisy labels. Thus, a question naturally comes up: can we find out the models which outperform the others during the training process?

With this question in mind, in this paper, we further explore the potential of rule-based noisy labels for overhead imagery segmentation. We show the memorization effect from both statistical and visual perspectives along with some other aspects of model learning behaviors. Based on these observations and inspired by [10], we propose a Peaks fusion assisted



(a) Optical image (b) Ground truth (c) Noisy labels

Fig. 2: One example of data triple used in this work. In (b) and (c), green, yellow, and dark blue represent trees, buildings, and background/others, respectively.

TABLE I: Quality assessment of rule-based noisy labels, where the uncertainty marked in brackets was estimated according to sets of 200 patches randomly picked. The overall of IoU and precision correspond to mIoU (mean Intersection of Union) and OA (Overall Accuracy), respectively.

class	trees	buildings	background	Overall
IoU	0.49(1)	0.46(2)	0.70(1)	0.55(1)
precision	0.60(1)	0.66(1)	0.93(1)	0.77(1)

EArly-Stopping (PEAS) approach to cope with the overfitting problem on rule-based noisy labels. First, a parametric curve fitting strategy is exploited aimed at determining the early-stopping point before models begin to memorize noise details. Next, a peaks fusion method using validation accuracies as indicators is included to improve the robustness of such early stopping strategy. The experiments show that our PEAS method can effectively prevent the models from falling into traps of noisy labels.

The rest of the paper is organized as follows. Sec. II describes the data we used in this work along with a short introduction of how the rule-based noisy labels were generated. After that, the proposed PEAS method is elaborated in Sec. III, followed by experimental results and conclusions in Secs. IV and V.

II. Dataset

The dataset used in this work is the 2017 New York City (NYC) dataset collected over the southwest area of New York City in the year of 2017. This dataset consists of three kinds of data, that is, multispectral orthophotos serving as model inputs, ground truth labels for model evaluation, and rule-based noisy labels collected from LiDAR data for model training. We first briefly introduce the automatic label generation system - *AutoGeoLabel* in Sec. II-A, and then summarize the dataset details in Sec. II-B.

A. Rule-based noisy label generation

LiDAR data contains rich information of land surface. However, handling 3D point cloud data is out of reach for many people due to the complexity of data characteristics and techniques. To avoid manipulation of 3-D data, *AutoGeoLabel* system rasterized the raw LiDAR data into a series of 2-D statistical feature layers using a sliding circle of 1.5m diameter. Within each circle, some basic statistics are calculated in terms of each quantity including elevation, counts of reflected pulses, and reflected pulse intensity. After that, noisy labels for trees and buildings were generated via combinations of binary classification formulas defined according to some simple rules. For instance, trees are expected to reflect the laser pulse multiple times leading to a high variation of counts, while rooftops of buildings are mostly flat with a near-zero standard deviation value of elevations. More technical details can be found in [8].

B. NYC dataset

The details of the three parts of the data are listed below:

- **Multispectral orthophotos** were obtained from the National Agriculture Imagery Program (NAIP) [12]. Each image patch contains 4 bands including near-infrared (NIR), red (R), green (G), and blue (B), with a spatial resolution of 1 meter.
- **Ground-truth labels** were gathered on basis of the 2017 LiDAR data with the aid of other data sources like vector GIS datasets [11]. This land cover layer originally contains 8 classes in total. To coordinate with the labeling system of rule-based noisy labels, only two classes, i.e., *trees* and *buildings* are considered. The rest 6 classes are merged into a general class - *background/others*. Such accurate land cover information is difficult to acquire in reality. Here we only use them for model evaluation purposes.
- **Rule-based noisy labels** were generated from LiDAR data via *AutoGeoLabel* system (cf. Sec. II-A). The used LiDAR data was collected from the project funded by Federal Disaster Recovery Community Development Block Grant ("CDBG-DR") for disaster recovery and resiliency initiatives of Superstorm Sandy [5]. The resolution of the 3-D point cloud data is approx. 10 points per square meter. One example of the automatically generated labels is shown in Fig. 2c, along with a statistical evaluation in Table. I.

To feed to model, all the data was clipped into small patches of 256×256 pixels. Ultimately, there are totally 6650 patch triples composed of orthophotos, ground truth maps, and noisy label maps.

III. Methodology

In this section, we present the details of the proposed PEAS method.

A. Early-stopping criterion

As mentioned above, similar memorization effects initially found on classification tasks also happen on segmentation tasks. The whole learning process thus can be divided into two stages, namely, the early learning stage and the noise pattern memorization stage. Noisy labels begin to explicitly undermine the model performance when the training process enters the latter stage. Therefore, detecting the turning point

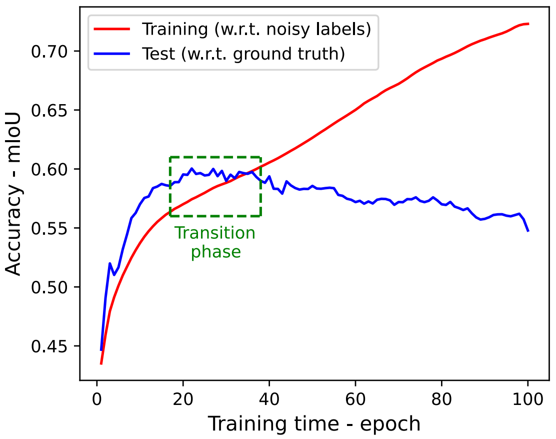


Fig. 3: Training and test accuracies (mIoU) versus training time (epoch), where the dashed green box indicates the transition phase from early learning stage to noise pattern memorization stage. Within the box, test accuracies start to decrease, while the growth of training accuracies gets slow. For better visual effects, curves shown here were smoothed using Savitzky-Golay filter with a window size of 3.

between two stages is crucial for early-stopping criterion design.

Inspired by [10], we utilize a fitting strategy to locate the endpoint of the first early learning stage. As shown in Fig. 3, when the model performance starts to degrade in test mIoUs (between model predictions and ground truth on test set), the growth of training mIoUs (between model predictions and noisy labels on training set) correspondingly slows down. Thus, we can promptly stop the training by setting a watchdog on the growth rate of training mIoUs. To this end, at the n-th epoch, we employ the least squares to fit the n training mIoUs at hand onto the following exponential parametric function with a, b, c being fitting parameters:

$$f(x) = a \cdot (1 - e^{-b \cdot x^c}), \tag{1}$$

the gradient of which, related to growth rate, is

$$f'(x) = abc \cdot e^{-b \cdot x^c} \cdot x^{c-1}, \tag{2}$$

where $0 < a \leq 1$ decides the magnitude of $f(x)$, $0 < c < 1$ with $c - 1 < 0$ ensures $f'(x)$ monotonically decreases, and $b > 0$ controls the curvature of the fitting line. In the fitting, the independent variable $x = 1, \cdots, n$ is the epoch indexes, while the dependent variable $f(x)$ is the corresponding training mIoUs at each epoch. After fitting, we can measure the deceleration by comparing the gradients of each epoch to that of the first one (also the biggest one) by

$$g(x) = \frac{q - f'(x)}{q} \tag{3}$$

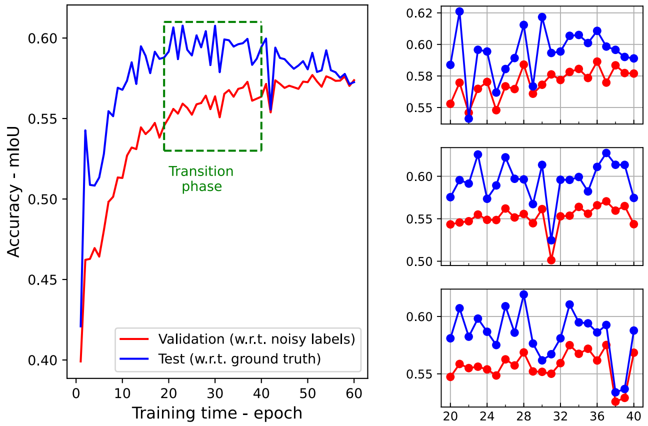


Fig. 4: Validation and test accuracies (mIoU) versus training time (epoch). Left: averaged results from 5 repeated experiments with different random splits of training and test samples. Right: zoomed landscapes in the transition phase from 3 single repeated experiments out of 5. In order to keep original tendencies of statistics, no smoothing filter was applied.

with the constant of $q = f'(1) = abc \cdot e^{-b}$. The early stopping is triggered when

$$e = \sum_{x=1}^{n} \text{sgn}\big[g(x)\big] < n \tag{4}$$

with

$$\text{sgn}\big[g(x)\big] = \begin{cases} 1 & \text{if } \ g(x) < r, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

where r is the predefined threshold of deceleration. Should the training terminates once Eq. (4) is true. In this case, e is the detected endpoint of early learning stage (it might be equal to or a bit smaller than the early-stopping point n), around which the peaks fusion strategy described later is applied. Otherwise, training continues.

Here we apply the fitting strategy to training mIoUs instead of class-wise training IoUs as claimed in [10]. The reasons are twofold. First, the aim of this work is early stopping. Relatively speaking, mIoUs give a better overall assessment of model performance on each class, which is more compatible to our goal. Next, training mIoUs read analogous functions as IoUs in [10] as illustrated in Fig. 3.

B. *Peaks fusion strategy*

Although the early-stopping strategy in Sec. III-A can avoid networks from overfitting to noisy patterns to some extent, it cannot guarantee the selected model fully explores the potential of noisy labels within the training process. As observed in Fig. 4, model performances fluctuate severely in the transition phase. The early-stopping criterion can only roughly identify the range where the best models might sit. To address this problem, we design a peaks fusion strategy to improve the robustness of this early-stopping method. Peaks here indicates that we select reliable candidates for model prediction fusion

TABLE II: Test accuracies obtained by different single models or fusion strategies, where our proposed PEAS method (tested with two m, i.e., the number of selected candidates/Cand. for fusion) is marked in bold, and the best and the 2nd best results are highlighted in red and blue colors, respectively. Specifically, $2b+1=m$ listed in *Fusion* part corresponds to the cases where all the models in the buffer zone are used for fusion. *Single model* includes three cases where models are fully trained on noisy labels, derived from the endpoints detected by our early-stopping criterion, and best-performed in terms of test mIoU during the training.

	buffer radius (b)	buffer size ($2b+1$)	#selected Cand. (m)	Test IoUs			
				Trees	Buildings	Others	mIoU
Fusion	1	3	3	0.519(19)	0.570(18)	0.713(08)	0.601(12)
	2	5	5	0.531(18)	0.582(13)	0.717(05)	0.610(10)
	5	11	11	0.539(12)	0.588(10)	0.719(02)	0.615(05)
	10	21	21	0.542(12)	0.594(11)	0.721(03)	0.619(07)
	10	**21**	**3**	**0.555(18)**	**0.592(13)**	**0.725(06)**	**0.624(11)**
	10	**21**	**5**	**0.551(16)**	**0.601(15)**	**0.726(05)**	**0.626(10)**
Single model	Fully trained on noisy labels			0.490(08)	0.451(19)	0.696(09)	0.545(11)
	Detected by early-stopping criterion			0.527(28)	0.560(39)	0.712(09)	0.600(21)
	Best during training			0.567(11)	0.591(07)	0.723(05)	0.627(08)

via the ”peaks” in validation mIoU landscapes. Note that validation mIoUs are calculated between predictions and noisy labels on validation set, since no ground truth labels are available during the training.

To select model candidates, a buffer zone is required. Let b denote the radius of the buffer zone centered on e. Then all the models falling into the buffer zone $[e-b, e+b]$ would be considered as choosing candidates. In this case, the real early-stopping point is supposed to be $e+b$ instead of e.

To finalize the predictions, a straightforward way is to average all the softmax outputs from $2b+1$ candidates. However, there are two drawbacks. To ensure the inclusion of enough well-performed models within the buffer zone, b cannot be set too small. The test time will greatly increase as b becomes larger. Moreover, those badly-performed candidates very likely impose negative impacts on the final fusion results. In this work, we select models guided by validation mIoUs. As can be seen from Fig. 4, the fluctuation tendencies of validation mIoUs (w.r.t. noisy labels) and test mIoUs (w.r.t. ground truth) are similar to each other in early learning stage. Our assumption is that severe fluctuations are mainly caused by clear structure information adjustment. Thus, mIoUs w.r.t. both ground truth and noisy labels would be affected. So we select m models associated with m peaks validation mIoUs within the buffer zone as follows,

$$\mathcal{F} = \text{argsort}_i(V_i, \text{order=descend})[:m], \tag{6}$$

where V_i is the validation mIoU at the i-th epoch with $i = \{e-b, \cdots, e+b\}$, $\mathcal{F}$ is the model index set recording the selected model candidates for fusion.

The final fusion results are derived from averaged predicted probabilities softmax$_{\mathcal{F}}$ on softmax outputs by models in $\mathcal{F}$, that is,

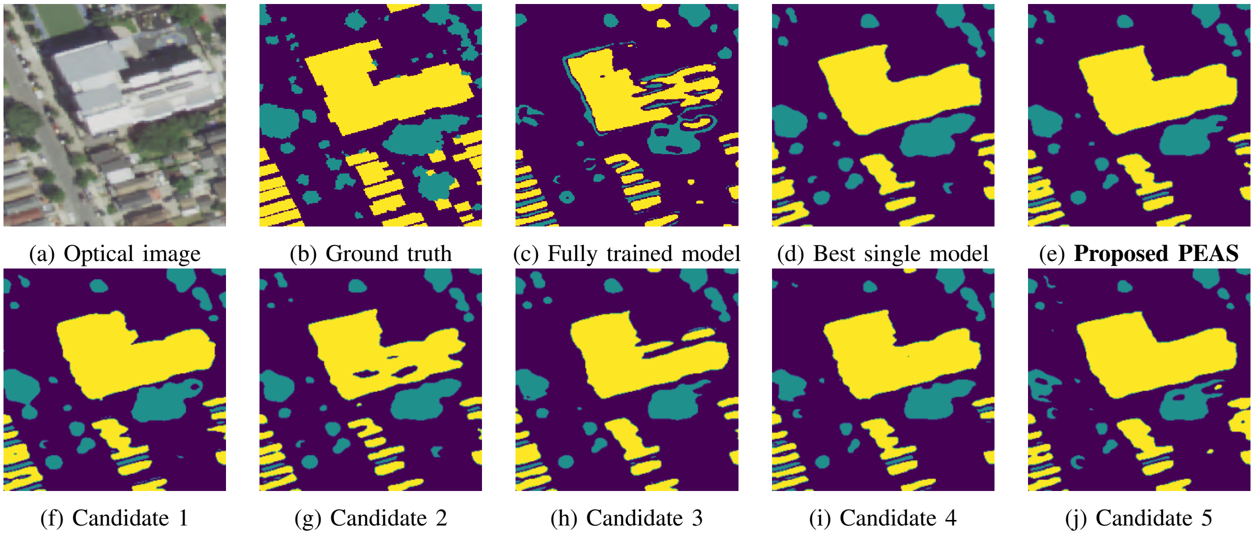
$$\text{softmax}_{\mathcal{F}} = \frac{1}{m} \cdot \sum_{i \in \mathcal{F}} \text{softmax}_i. \tag{7}$$

TABLE III: Test accuracies obtained from different replays with buffer radius $b=10$, number of selected candidates from the buffer zone $m=5$, where *detect* indicates from which epochs models are derived. Also, statistics in red and blue are the best and the 2nd best results in each replay.

Replay (detect)	Cand. (detect)	Test IoUs				Best (detect)
		Trees	Buildings	Others	mIoU	
1 (30)	36	0.536	0.577	0.720	0.611	0.626 (21)
	28	0.571	0.553	0.723	0.615	
	38	0.530	0.543	0.715	0.596	
	34	0.513	0.591	0.718	0.607	
	33	0.526	0.576	0.719	0.607	
	PEAS	**0.542**	**0.596**	**0.725**	**0.621**	
2 (30)	37	0.569	0.590	0.724	0.627	0.627 (37)
	36	0.514	0.601	0.719	0.611	
	39	0.582	0.533	0.726	0.613	
	34	0.517	0.566	0.715	0.599	
	26	0.554	0.588	0.724	0.622	
	PEAS	**0.564**	**0.618**	**0.732**	**0.638**	
3 (32)	40	0.584	0.604	0.730	0.639	0.639 (40)
	35	0.585	0.565	0.721	0.624	
	37	0.525	0.546	0.709	0.593	
	41	0.557	0.569	0.722	0.616	
	31	0.555	0.580	0.717	0.618	
	PEAS	**0.576**	**0.611**	**0.730**	**0.639**	
4 (38)	44	0.531	0.554	0.718	0.601	0.623 (22)
	37	0.525	0.523	0.710	0.586	
	43	0.540	0.552	0.718	0.603	
	48	0.514	0.521	0.705	0.580	
	39	0.546	0.516	0.706	0.589	
	PEAS	**0.541**	**0.574**	**0.722**	**0.612**	
5 (31)	41	0.520	0.563	0.707	0.597	0.619 (28)
	37	0.507	0.565	0.706	0.593	
	33	0.547	0.567	0.717	0.610	
	35	0.497	0.579	0.706	0.594	
	28	0.549	0.593	0.716	0.619	
	PEAS	**0.532**	**0.606**	**0.718**	**0.619**	

IV. Experiments

In this section, ahead of describing detailed experimental results in Sec. IV-B, we first present our settings in Sec. IV-A for reproduction purposes.


(a) Optical image (b) Ground truth (c) Fully trained model (d) Best single model (e) **Proposed PEAS**

(f) Candidate 1 (g) Candidate 2 (h) Candidate 3 (i) Candidate 4 (j) Candidate 5

Fig. 5: Examples of segmentation maps along with corresponding optical image and ground truth. Specifically, (f)-(j) are maps produced by 5 candidate models used to generate final fusion prediction shown in (e).

A. Settings

The settings are summarized as follows:

- **Network architecture**: We utilized the vanilla U-Net [3] as our segmentation model, which is composed of 4 downsampling modules and 4 upsampling modules in sequence. In each down/upsampling module, the combination of a convolutional layer with kernel size of 3, a batch normalization layer, and a ReLU layer is repeated twice after down/upsampling.
- **Training and test sets**: The whole dataset (6650 image triples in total) was randomly split into two subsets with 5600 for training and 1050 for test. In the training set, 10% samples were further selected as the validation set. To measure the uncertainty, the random split was repeated 5 times to retrain the model. Hereinafter, the uncertainty is shown in brackets, and the results obtained with different splits are denoted as Replay 1-5.
- **Training**: Adam optimizer is adopted in our experiments with an initial learning rate of 1e-3 and a weight decay of 1e-8. The batch size is set to 50. The loss is a joint one of the cross entropy loss and the dice loss for optimization. Also, we continue the training till 100 epochs after successfully detecting the early-stopping point, in order to obtain the well-trained models on noisy labels for comparison.
- **PEAS settings**: The early stopping threshold r is set to 0.97 via cross validation. The buffer radius and the number of selected candidates are empirically set to 10 and 5, respectively.

B. Results

First, TABLE II lists the test accuracies calculated on ground truth data averaged from 5 replays. Except for PEAS, results obtained by fusion without peaks selection, and some single models are also presented in TABLE II. Notice that ”Best during training” refers to the highest test mIoUs among all the epochs. Such model is nearly impossible to identify in practice. We include those results for comparison purposes only, since it can give us a rough overview about the potential of models purely trained with noisy labels. It can be found from TABLE II that our proposed method can get comparable results to what the best model can achieve during the training. Without early stopping, model would finally overfit to noisy patterns resulting in inferior performances. Without peaks fusion strategy, the models solely detected by early stopping criterion are unstable, featured with high standard deviations. Moreover, fusing all the models lying within the buffer zone would only lead to marginal improvement in comparison to early-stopping criterion detected models yet with more computation required. But they fail to outperform PEAS partly due to the negative effects of some badly-performed fusion candidates.

Then, TABLE III gives a more detailed overview of the peaks fusion strategy in each replay. Similar to what we can concluded from TABLE II, in each replay, the proposed method can get close and sometimes even better results compared to the best single model results during the training. Two facts contribute to this phenomenon. In all the replays, the peaks selection strategy is able to filter out some extremely badly-performed models within the buffer zone, avoiding them greatly biasing the fusion process. Besides, fusion from multiple models can boost the robustness of model performance.

Finally, Fig. 5 shows an example of generated segmentation maps, from which we can conclude that the proposed method is capable to yield satisfying visual results by gathering information from multiple candidate models, while the map by the fully trained model is grandly hurt by noise patterns. It means that early stopping strategy is crucial to avoid model from overfitting to label noises.

V. Conclusion

This work proposed a new peaks fusion assisted early-stopping (termed as PEAS) strategy to fully exploit the potential of noisy labels by the rule-based automatic label generation system for overhead imagery segmentation. We found that the memorization effect also happens on segmentation tasks when the labels of training data are contaminated by noises. But the learning process starts to slow down when model begins to memorize noise details. Therefore, a fitting based early stopping criterion is designed to dynamically terminate the training. Besides, to reduce the negative effects of fluctuation during the training, a peaks fusion strategy is used to assist the early stopping method, which enhances the robustness of the proposed approach by selecting a few relatively reliable models to make final decisions. Our experiments verified the effectiveness of the proposed method. In the future, we plan to design some noise cleaning strategies on basis of early learning results to further improve model performance trained with noisy labels.

Acknowledgment

This work is supported 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)". The work of X. Zhu is additionally supported 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) and by German Federal Ministry for Economic Affairs and Climate Action in the framework of the "national center of excellence ML4Earth" (grant number: 50EE2201C).

References

[1] Stephan Pauleit and Friedrich Duhme. "Assessing the environmental performance of land cover types for urban planning". In: *Landscape and urban planning* 52.1 (2000), pp. 1–20.

[2] Chandra P Giri. *Remote sensing of land use and land cover: principles and applications*. CRC press, 2012.

[3] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[4] Devansh Arpit et al. "A closer look at memorization in deep networks". In: *International conference on machine learning*. PMLR. 2017, pp. 233–242.

[5] *New york city lidar.* 2017. URL: https://data.cityofnewyork.us/api/geospatial/uyj8-7rv5?method=export&format=Original.

[6] Xiao Xiang Zhu et al. "Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources". In: *IEEE Geoscience and Remote Sensing Magazine* 5.4 (2017), pp. 8–36. DOI: 10.1109/MGRS.2017.2762307.

[7] Mohammadreza Sheykhmousa et al. "Post-disaster recovery assessment with machine learning-derived land cover and land use information". In: *Remote sensing* 11.10 (2019), p. 1174.

[8] Conrad M Albrecht, Fernando Marianno, and Levente J Klein. "AutoGeoLabel: Automated Label Generation for Geospatial Machine Learning". In: *2021 IEEE International Conference on Big Data (Big Data)*. IEEE. 2021, pp. 1779–1786.

[9] Chiyuan Zhang et al. "Understanding deep learning (still) requires rethinking generalization". In: *Communications of the ACM* 64.3 (2021), pp. 107–115.

[10] Sheng Liu et al. "Adaptive early-learning correction for segmentation from noisy annotations". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 2606–2616.

[11] New York City. *Land Cover NYC, 2017*. URL: https://data.cityofnewyork.us/Environment/Land-Cover-Raster-Data-2017-6in-Resolution/he6d-2qns.

[12] *USGS EROS Archive - Aerial Photography - National Agriculture Imagery Program (NAIP)*. URL: https://www.usgs.gov/centers/eros/science/usgs-eros-archive-aerial-photography-national-agriculture-imagery-program-naip.