SyntCities: A Large Synthetic Remote Sensing Dataset for Disparity Estimation

Mario Fuentes Reyes, Pablo D'Angelo, and Friedrich Fraundorfer, *Member, IEEE*

***Abstract*—Studies in the last years have proved the outstanding performance of deep learning for computer vision tasks in the remote sensing field, such as disparity estimation. However, available datasets mostly focus on close-range applications like autonomous driving or robot manipulation. To reduce the domain gap while training we present SyntCities, a synthetic dataset resembling the aerial imagery on urban areas. The pipeline used to render the images is based on 3-D modeling, which helps to avoid acquisition costs, provides subpixel accurate dense ground truth and simulates different illumination conditions. The dataset additionally provides multiclass semantic maps and can be converted to point cloud format to benefit a wider research community. We focus on the task of disparity estimation and evaluate the performance of the traditional semiglobal matching and state-of-the-art architectures, trained with SyntCities and other datasets, on real aerial and satellite images. A comparison with the widely used SceneFlow dataset is also presented. Strategies using a mixture of both real and synthetic samples are studied as well. Results show significant improvements in terms of accuracy for the disparity maps.**

***Index Terms*—Disparity estimation, synthetic imagery, urban reconstruction.**

I. Introduction

ALGORITHMS for disparity estimation have been widely studied in the last decades in different fields, including the remote sensing community. It aims to find the correspondence between two or more rectified images and retrieve the shift for the pixels location along the epipolar line. From the disparities it is possible to compute depth values for the objects present in the samples, which helps to reconstruct 3-D scenes. Most of the traditional algorithms follow a pipeline with matching cost computation, cost aggregation, disparity estimation, and disparity refinement [1].

3-D reconstruction has also been studied by the remote sensing community, where the input images are processed to generate data, such as digital surface models (DSM). However, the nature of the remote sensing imagery is challenging for many stereo matching algorithms. Seasonal changes, atmospheric and illumination conditions, urban redevelopment, among others, modify the appearance and content of the captured scenes. Additional difficulties for a successful matching are imposed by the presence of texture-less, patterned, and non-Lambertian surfaces. What is more, the disparity range for high mountains or buildings varies significantly with respect to the one required for objects with low elevation.

While traditional approaches like semiglobal matching (SGM) [2] perform well to estimate disparities for many scenes, deep learning algorithms are now the state of the art, having a better generalization for complicated areas [3], [4].

Nevertheless, the improved performance offered by deep learning algorithms demands a large amount of samples for training, which is sometimes limited or incomplete in remote sensing. Due to its nature, aerial/satellite-borne data are expensive and its acquisition requires planning to avoid bad weather conditions. Also, the ground truth for disparity estimation is usually obtained from LiDAR, that produces a sparse result and makes it difficult to define sharp boundaries or detect small objects. In addition, LiDAR shows different behavior in vegetated areas, especially trees, and needs to be captured simultaneously to avoid systematic differences due to scene changes, such as vegetation growth and building activities. 4-D light fields and plenoptic cameras are also a resource to generate high quality 3-D models [5], but this technology cannot be used during aerial and satellite data acquisition.

Taking into account the difficulties mentioned above, we propose a new synthetic dataset for disparity estimation. Since the rendering is obtained via software, we are able to generate dense ground truths with sharp boundaries and subpixel accuracy. In addition, we simulate different illumination conditions, ground sample distances, and baselines for the stereo system. One of the novelties of the proposed dataset is its remote sensing oriented application by using models that resemble urban areas to reduce the domain gap.

We train different state-of-the-art networks on our generated samples and test the models on real satellite and airborne data. Besides, we compare the results by training with the widely used SceneFlow [6] dataset, where the disparity maps are oriented on close-range applications.

Our main contributions in this article are the following.

1) We present SyntCities, the first (to the best of our knowledge) large synthetic dataset to train disparity estimation

Manuscript received 30 May 2022; revised 16 August 2022 and 13 October 2022; accepted 15 November 2022. Date of publication 23 November 2022; date of current version 30 November 2022. The work of M. F. Reyes is currently funded by a DAAD-DLR Research Fellowship 57478193 to pursue his Ph.D. studies. *(Corresponding author: Mario Fuentes Reyes.)*

Mario Fuentes Reyes and Pablo D'Angelo are with the Department of Photogrammetry and Image Analysis, Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), 82234 Wessling, Germany (e-mail: mario.fuentesreyes@dlr.de; pablo.angelo@dlr.de).

Friedrich Fraundorfer is with the Department of Photogrammetry and Image Analysis, Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), 82234 Wessling, Germany, and also with the Institute of Computer Graphics and Vision, Graz University of Technology, 8010 Graz, Austria (e-mail: fraundorfer@icg.tugraz.at).

SyntCities can be downloaded at: https://tinyurl.com/77e3n6m9.

Digital Object Identifier 10.1109/JSTARS.2022.3223937

focused on remote sensing imagery. Ground truth maps are dense and offer subpixel accuracy.
2) We conducted a set of experiments on recent neural networks to analyze the advantage of performing data augmentation with our generated samples.
3) By comparing with other datasets, we reduce the estimation error and improve the 1-pixel accuracy, which is of crucial importance for the generation of DSMs.
4) We show how SyntCities has good generalization capabilities to be used even on unseen data for inference of disparity maps.
5) We share the data in formats that can be further processed (like point for cloud generation) and include multiclass semantic maps.

II. Related Work

In this section, we discuss first the existing work oriented to the generation of synthetic datasets, its applications and limitations. Second, we mention some studies related to possible usage of both disparity estimation and semantics segmentation, since we provide these maps in our dataset and might encourage the research community to conduct further experiments in this direction. For our own experiments, we focus on the disparity estimation part.

A. Synthetic Datasets

Deep learning has helped to outperform many algorithms related to computer vision recently, but it also demands a large amount of data to train models that can generalize for testing images from different sources. However, such large amount of information is not always available or is expensive to acquire. Therefore, the application of synthetic datasets is an option that can compensate the lack of real data for the training process. In many cases, these datasets are used for pretraining stages and smaller sets of real data are applied to finetune the models and reduce the domain gap.

One of the first available synthetic options was the MPI Sintel Dataset [7], where frames are taken from an open source movie and rendered to evaluate optical flow algorithms. The samples were extended to facilitate other tasks, such as semantic segmentation, camera motion, and stereo matching. In the same way, the SceneFlow dataset [6] was proposed to train neural networks for optical flow, but increasing the number of samples to 34 K (instead of 1 K as Sintel). Due to its large size and variety of objects and textures, SceneFlow has been one of the main references to pretrain networks for different tasks. It includes scenes from a movie, random objects, and resembling a car perspective on the streets.

Autonomous driving has also benefited from the synthetic imagery. While real images are part of available datasets, these are limited in size and might lead to the overfitting of the models. The KITTI 2012 [8] and KITTI 2015 [9] datasets include images from cameras with a driver's perspective, where elements like streets, cars, houses, or vegetation are part of the scene. They also include a ground truth from a laser scanner, providing accurate values for depth. In addition, files for odometry or semantics ease their application for other tasks. However, the number of samples (around 400 pairs) limits its implementation for deep learning architectures and the sparse measures from the depth sensor provide an incomplete disparity map. As a feasible solution to balance the amount of required data, SceneFlow can be used to pretrain the models for disparity estimation, while the SYNTHIA dataset [10] is a suitable option for the semantic part. SYNTHIA also focuses on autonomous driving and is similar in terms of content and geometry to the KITTI datasets. In contrast, it consists of more than 13 K samples and dense ground truth maps. Another similar approach is the ParallelEye dataset [11] based on a pipeline of the CityEngine and Unity3D software suites. It also includes information for object detection and tracking.

Nonetheless, the alternatives described above are oriented to close range applications, which is not suitable for remote sensing, where large areas are covered and small errors in the disparity estimation lead to significant inaccuracies in the DSMs. The Urban Semantic 3-D (US3D) dataset [12] was proposed for the Data Fusion Contest 2019 (referenced as "grss_dfc_2019" for the contest itself, but we keep it as US3D in the current article) and included a stereo matching track. Although the number of samples enables the training of deep learning architectures, the disparity maps are not complete (with a default value assigned to many pixels) and do not archive subpixel accuracy, which imposes a significant error when computing the depth. In addition, a multiyear difference between image and ground truth LiDAR acquisition causes many inconsistencies due to vegetation, building, and infrastructure changes. Using multidate imagery also affects the vegetation measurements, since it has visible seasonal changes in terms of color and density. Despite the fact that training with this data might affect the performance of the networks, testing on such imagery is still one of the few options for real large areas.

Developing synthetic datasets within the remote sensing environment has also been studied, although only few publications address it. The WHU dataset [13] is based on real aerial images and then merged on a DSM. After that, images are rendered via software from the generated DSM and it produces a synthetic output in form of disparity maps. Ground truth is obtained as dense maps, but the accuracy of the DSM is constrained by the algorithms of the ContextCapture software.

Under these circumstances, we have developed a new synthetic dataset. Considering that the 3-D software has detailed information of the geometric content of the scenes, dense and accurate ground truths can be achieved. Furthermore, expanding the dataset for additional views or different simulated conditions can be easily done, reducing costs and time.

B. Approaches to use Both Disparity and Semantic Maps

Although the present work focuses on the disparity estimation, the provided semantic maps can be a helpful resource for research making use of both data sources, since these exploit the geometric information from the scene. This idea was recently addressed on the Data Fusion Contest 2019 [14], [15], [16], where semantic and disparity maps are predicted and evaluated for the same regions on one of the tracks.

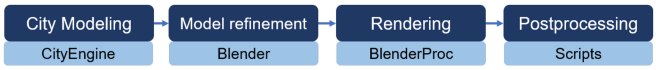

Fig. 1. Simplified pipeline used for the proposed dataset generation.

Real datasets for semantic segmentation, such as US3D have incomplete semantic maps, with noisy buildings and many elements without an assigned category. On the contrary, synthetic datasets avoid expensive manual annotations and provide sharp dense maps. An existing synthetic example is the Synthinel-1 dataset [17], where models from the CityEngine software are rendered to create segmentation maps with the labels building/no-building. While the pipeline is an efficient way to generate the data and resembles real imagery, the ground truth is limited to two classes and depth information is not included.

Some publications have already studied the usage of both input sources. In SegStereo [18] the semantic information is embedded in the network and also being learned as an intermediate step to refine the disparity map. GIO-Ada [19] learns to reduce the domain gap by creating intermediate samples with a more realistic appearance and later estimates both semantic and depth maps. DispSegNet [20] proposed an architecture similar to SegStereo but using the semantic embedding for the disparity loss and created an enhanced cost volume to improve the accuracy. RTS^{2}Net [21] focused on real-time efficiency and followed a coarse to fine design. SSPCV-Net [22] considered pyramid cost volumes to describe semantics and geometry. CorDA [23] used the depth estimation as an intermediate step to retrieve the disparity maps and with this information reduced the domain gap.

Many of these methods achieve good quality results, but at least for the disparity estimation, they do not compete with the state-of-the-art solutions in terms of accuracy. By releasing this dataset, we intent to facilitate further research in this direction.

III. Dataset Generation and Description

The generation of the dataset makes use of different 3-D software suites and scripts for modeling, rendering, and postprocessing. In Fig. 1, a simplified description of the adopted pipeline is shown. The detailed steps are explained in the following paragraphs.

A. City Modeling

CityEngine is a software that allows to build cities in a 3-D environment and follows the CGA shape grammar language. Large models can be created from Open Street Map (OSM) and user defined rules for the city architecture and its distribution. In the current article, we started from the example models for New York, Paris, and Venice that are publicly available on the Esri platform.

Empty areas from the examples were replaced with parks and buildings to set content in all the regions of the scene. Vegetation was changed to textured ellipsoid models instead of the intersected planes to have a more natural distribution of depth values. In addition, we used the script option within the CityEngine environment to separate the buildings according to the rooftop type, this is done to provide the additional semantic maps.

A model including only the buildings belonging to each roof type and a full model including all elements in the scene are exported. All cases were exported in Wavefront (.obj) format. CityEngine consumes approximately 17 GB of RAM memory to manipulate the full models, and requires few minutes to export the whole scene.

B. Model Refinement

The models were later imported in the Blender software, which is an open source for 3-D modeling, animation, and rendering. Here, the objects were split into different categories, which are represented in the ground truth segmentation maps. The objects were created by separating the faces of the complete 3-D scene according to the image file used as texture. This does not apply to the buildings, which were previously separated by roof type in CityEngine. A single file in COLLADA (.dae) format is exported with the merging of all objects.

Illumination conditions and camera properties are studied in the 3-D environment to set the appropriate values for each city. The light is set to the Sun mode to have a homogeneous brightness in the whole area. A vertex located close to the center of each model is used as a reference to set the camera positions. Apart from that, changes are applied on the reflection properties of the surfaces as well as on the noise distribution for textures. Minor editing was also conducted to avoid empty regions that might lead to the presence of outliers. Furthermore, we set a 3-D plane below the models as background, which avoids infinite depth while rendering for not defined regions. The manipulation and edition of the models in Blender requires approximately 8 GB of RAM memory.

C. Rendering

Once the models were complete, we utilized the BlenderProc pipeline [24] to render within the Blender environment. BlenderProc requires a detailed configuration file to set properties, such as camera positions, camera parameters, stereo configuration, illumination conditions, output resolution, etc.

Our approach wraps BlenderProc, so we can define externally the main parameters for our dataset. Here we also set the camera positions according to the size of the city model and the desired overlapping between samples. The stereo rig configuration is computed from the base-to-height ratio and allows different baselines. The configuration file required by BlenderProc is then built with the specified parameters.

In addition, we manipulated the antialiasing filters to produce smooth borders in the RGB samples but sharp edges for the depth maps. For each camera position we rendered a pair of RGB images, their depth maps and their segmentation maps.

We also experimented the option to produce instance maps (where each building would be assigned a label), but the computational cost is too high even for one camera position. Rendering a pair in instance segmentation mode requires around 200×

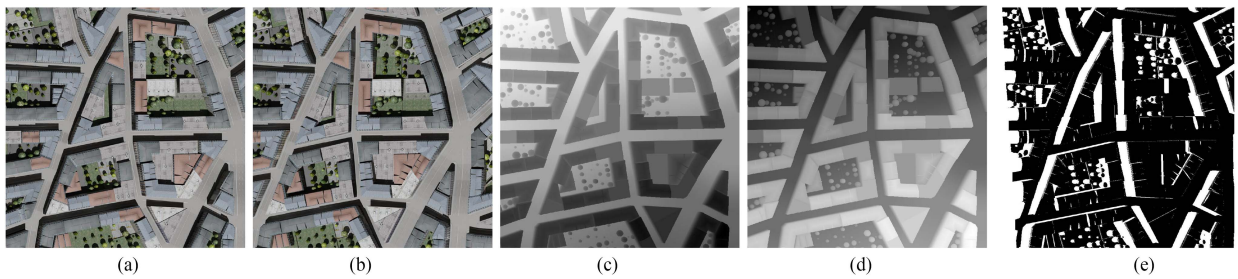

Fig. 2. Samples from the SyntCities dataset. Optical imagery used for input is shown in (a) for the left and (b) for the right view, with the respective depth and disparity maps for the left view in (c) and (d) (Samples for the right view are also available, but not shown in this image). In (e) we illustrate the left-right consistency masks, where the region in white is not visible in both views. (a) RGB - left view. (b) RGB - right view. (c) Depth map. (d) Disparity map. (e) Consistency mask.

longer than the semantic case. The rendering process for SyntCities takes a bit more than five days using a NVIDIA Quadro P1000 graphics card with 4 GB memory and Blender 2.93.

D. Postprocessing

RGB images and semantic maps were directly obtained from the rendering process. In contrast, the depth map has to be translated into a disparity map. Since the depth is measured in a radial way from the center of the camera, we transformed it into distance to the camera plane first. After that, the distance is used with the known camera parameters to compute the disparity as follows:

$$d = \frac{f \cdot b}{z} \tag{1}$$

where d is the disparity, f the focal length, b the baseline of the stereo rig, and z the distance to the plane. The disparity values are then transformed into pixels. As a result of the different baselines applied to create the dataset, occlusions are present in many samples. Therefore, we also created left-right check consistency maps to mask pixels that are not visible in both views. The threshold for consistency is set to 1 pixel.

Homogenization of categories between different models and rendering conditions is also applied, so the labels remain coherent between all the samples. For users requiring the camera extrinsic and intrinsic matrices, we also provide these in separate files for each camera position and view. Such matrices are usually expected for multiview stereo (MVS) neural networks.

E. Description

The presented dataset includes a total of 8100 image pairs with the following features.

1) Three city models: New York, Paris, and Venice.
2) Three ground sampling distances (GSDs): 10 cm, 30 cm, and 1 m.
3) Three azimuth angles (150°, 180°, and 210°) and three elevation angles (20°, 50°, and 70°) for the simulated Sun light.
4) Four base-to-height ratios (BH) per city: 0.1, 0.3, 0.5, and 0.9 for Paris and Venice; 0.03, 0.07, 0.10, and 0.12 for New York.
5) For each combination of the previous parameters 20 pairs are available for training and 5 for testing. This split is fixed for all cases.
6) Disparity values are mainly in the range of [0,192]. This facilitates its direct usage in deep learning frameworks, where the cost volumes usually use such range to estimate the disparities.

On the Fig. 2, we show samples from the dataset for a small region on the simulated Paris model. The 8100 pairs include a similar subset of images, camera parameters, and rendering conditions. All images have a resolution of 1024 × 1024 pixels.

F. Semantic Categories

As mentioned before, semantic maps are also included. There are 13 categories available: vegetation, streets, rooftops (mansard, gambrel, gable, hip and flat styles), facades, gardens, landmarks, cars, and background. Fig. 3 shows an example of the semantic maps for the same patches represented in Fig. 2. Samples for both left and right view are available.

G. Data for Point Cloud Generation

Taking advantage of the available rendered maps and known camera parameters in SyntCities, we explored the possibility of generating point clouds based on the depth and semantic maps. We utilized the Open3D library [25] for this purpose.

Due to their large file sizes, we do not include these outputs in the dataset, but this can be easily generated from the provided images.

Although we did not conduct any experiments in this direction, we consider this would be helpful for deep learning strategies applied to point clouds, specially because the type of rooftops and other geometries can be learned.

IV. Disparity Estimation Experiments

We have conducted a series of experiments to analyze the advantages of training architectures on SyntCities for the disparity estimation. Aside from our proposed dataset, we also worked with samples from SceneFlow, US3D, and an aerial 4 K dataset processed by DLR [26].

SceneFlow is the main reference to train networks for disparity estimation due to its large size, but as we have previously

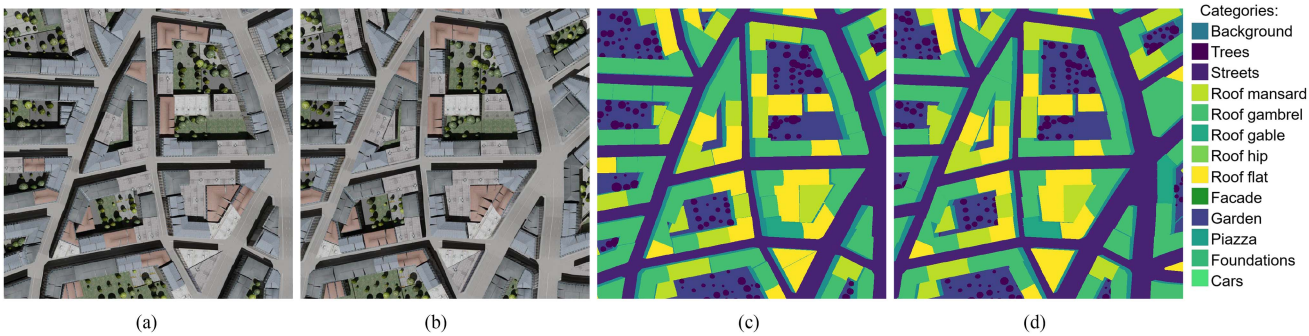

Fig. 3. Additional samples from SyntCities. Optical imagery used for input is shown in (a) for the left and (b) for the right view, with the respective segmentation maps in (c) and (d). Colors for each category are displayed in the list at the right. (a) RGB - left view. (b) RGB - right view. (c) Segmentation map left. (d) Segmentation map right.

mentioned it is oriented to close-range applications. Hence, we want to compare how networks perform while training with both synthetic options SceneFlow and SyntCities, to investigate if the domain gap with respect to real satellite/aerial imagery is reduced.

On the other hand, we also consider two real datasets. First, we take samples from US3D covering areas above Jacksonville, Florida and Omaha, Nebraska. The images are captured by the WorldView3 satellite with 30-cm GSD for the panchromatic case. The ground truth is obtained from an aerial LiDAR sensor and almost 4000 pairs are available for training.

Second, we use a 4 K collection of aerial imagery covering the area of Gilching, Germany with 6.9-cm GSD. The reference disparity map for these samples is obtained by an SGM implementation for multiview stereo matching, where a high-quality DSM is cropped to match the location of the images. Because of the size of this dataset (we consider only 16 images, where urban and semiurban areas are covered), we use the samples only to test the algorithms.

A. *Stereo Matching Algorithms*

SGM has been the main algorithm for stereo matching in the last decades. Its compromise between accuracy and computational cost makes it a feasible option for many applications and is used in open source pipelines for 3-D reconstruction like S2P [27]. Unlike deep learning architectures, SGM does not need to be trained on the target domain. Nevertheless, the computation of the aggregated cost requires parameters that are set empirically and have to be adapted to the features of the input images. Those parameters limit the performance of the algorithm and might lead to incomplete disparity maps as outputs.

Deep learning approaches on the other hand require large volumes of data. Even when recent state-of-the-art architectures outperform SGM and traditional methods, the models are not able to handle easily changes in the target domain. For example, a network that has been trained on data for autonomous driving might have a poor performance when applied for remote sensing imagery. Moreover, the training process frequently takes days and a high computational cost in terms of memory and GPU usage.

Despite the drawbacks mentioned above for deep learning, it performs better than traditional algorithms having enough data and a reliable ground truth. Since the publication of MC-CNN [3], where a cost volume is generated with convolutional neural networks, many architectures have achieved outstanding performance for benchmarks like KITTI or Middlebury [1].

Some other remarkable approaches include the first end-to-end architectures Disp-Net [6] and GC-Net [28], where post-processing steps, such as SGM are removed and the refinement of the disparity maps is embedded in the learning process. A significant improvement was later presented with the design of PSMNet [29], which includes a pyramid pooling model to recover more context information and makes use of 3-D convolutions to regularize the cost model, a strategy used in many further architectures. Based on a similar principle to SGM, GANet [4] evaluates the costs along different directions to refine the cost volume and avoid discontinuities. To reduce the domain gap presented in the previous networks, DSMNet [30] applies a domain-invariant normalization which benefits of the synthetic imagery. Nevertheless, its performance is not as good as GANet when using the same training dataset. A different concept is presented in AANet [31] to reduce both memory consumption and inference times, while slightly decreasing the accuracy. More recently, strategies consisting of gated recurrent units (GRUs) have been introduced to computer vision tasks with an outstanding performance. This has been applied to the disparity estimation problem, where RAFT-Stereo [32] includes a series of GRUs to estimate maps at full resolution and with high accuracy. In a different strategy, SMAR-Net [33] includes a GAN to compensate for sparse ground truths by warping the left image with the disparity map.

For this article, we train our models in two networks: GANet and AANet. The reason to select these networks is the accuracy for GANet and the reduced computational cost of AANet, being both also a common framework to compare other architectures.

GANet includes two types of novel layers named semiglobal guided aggregation (SGA) and local guided aggregation (LGA). SGA is based on a principle similar to SGM by considering four directions for the cost aggregation step and LGA recovers information from thin structures. The parameters that are empirically

TABLE I
COMPOSITION OF THE INPUT DATA FOR THE PROPOSED EXPERIMENTS WITH GANET

	Datasets		
Training model	**SF**	**SC**	**US3D**
GA-SF	100	0	0
GA-SC	0	100	0
GA-SCd	0	100	0
GA-US3D	0	0	100
GA-95SC	0	95	5

The GA-SCd case corresponds to the "deeper" version in the GANet paper. Values are expressed as percentages.

TABLE II
COMPOSITION OF THE INPUT DATA FOR THE PROPOSED EXPERIMENTS WITH AANET

	Datasets		
Training model	**SF**	**SC**	**US3D**
AA-SF	100	0	0
AA-SC	0	100	0
AA-US3D	0	0	100
AA-80SF	80	0	20
AA-80SC	0	80	20
AA-95SF	95	0	5
AA-95SC	0	95	5
AA-99SF	99	0	1
AA-99SC	0	99	1

Values are expressed as percentages.

set in SGM are adapted in the model to be learned while training. GANet outperformed the PSMNet (which had the best result for KITTI back then) and generates accurate results on subpixel level. However, the training process might take many days and is computationally demanding.

To reduce the memory and time consumption we conduct experiments with AANet as well. AANet introduces two adaptive aggregation approaches in an intra- and cross-scale manner. The intra-scale aggregation is similar to deformable convolution [34], [35] and adds an offset to the convolutional filters to improve the quality of the result around boundaries and thin structures. The cross-scale aggregation shares information between different scales. Its based on the idea that correspondences in the coarsest scale are more discriminative in textureless regions and this can guide the algorithm in the finer scales.

B. GANet Experiments

We trained the GANet network with different samples and tested on real aerial and satellite data. The configurations for training are listed in Table I. For each training, we show the percentage of each available dataset that was used as input. From this point on, we use SF and SC as acronyms for SceneFlow and SyntCities, respectively, specially to describe the experiments and results based on this data.

The SceneFlow model was trained only for 10 epochs due to its very large size (more than 35 K pairs are included) and took more than six days. For the other cases we trained for 27 epochs, resulting on two days of training time and four days in the GA-SCd case. GA-SCd corresponds to the "GANet deep" model presented by the authors in the original paper and includes more layers than the basic model. Here, 6480 image pairs are taken as input, corresponding to all the training samples (80% out of the 8100 available). For the GA-95SC instance, we want to observe the performance of the training when a real but small dataset is available and we can mix the samples with the synthetic ones to compensate the lack of data. We used 4750 samples from SyntCities and 250 from US3D. The GA-US3D model had 4000 samples for training.

Training was conducted on four GeForceRTX 2080 GPUs with 12 GB memory each, a batch size of 4, patches with 432×432 pixels size, a disparity range of $[0, 192]$, and the other parameters have the default values of the GANet implementation.

C. AANet Experiments

Similarly, we trained AANet with different configurations. Because of the reduced memory consumption and faster training, we conducted an extensive set of experiments. Table II shows the configurations for the different training models, following the same description system, as explained for Table I. The AA-SF model was trained for 64 epochs, as suggested in the AANet paper. For the other models we adapted accordingly the number of epochs to have a similar training time (around 48 h each). AA-SF is trained again with more than 35 K pairs, AA-SC is trained with 6480 pairs for 350 epochs, AA-US3D with 4000 pairs for 560 epochs and the other models with 5000 samples for 450 epochs. Many cases with mixed sources are trained to observe the advantages of data augmentation from synthetic imagery. For all these options we used both SceneFlow and Syntcities. Again, we trained on four GeForceRTX 2080 GPUs with 12 GB memory each, a batch size of 24, patches with 288×576 patch size, and a disparity range of $[0, 192]$. Other parameters are kept with the default values.

V. DISPARITY ESTIMATION RESULTS

The trained models were tested on the US3D and the aerial 4 K datasets. We evaluated four metrics to assess the quality of the results. For the statistical metrics, we use the median-based values instead of the mean-based ones because of their robustness to outliers and their capabilities to summarize skew distributions better [36]. First, we compute the median of the difference between the ground truth and the generated disparity maps. For this metric we did not consider the median of the absolute difference to use it as an indicator of a possible bias. This is computed as

$$\text{Median}_{\text{diff}} = \text{median}(X_{\text{diff}}), \quad X_{\text{diff}} = X - \bar{X} \tag{2}$$

where X is the ground truth, $\bar{X}$ is the generated result, and X_{diff} is the difference between both. Second, we compute the median absolute deviation (MAD) of the difference as

$$\text{MAD}_{\text{diff}} = \text{median}(\mid X_{\text{diff}} - \widetilde{X}_{\text{diff}} \mid), \quad \widetilde{X}_{\text{diff}} = \text{median}(X_{\text{diff}}). \tag{3}$$

The absolute value is used in this occasion to analyze the precision of the disparity values. We also consider the 3 pixel

TABLE III
RESULTS OF GANET FOR THE US3D DATASET

	Algorithms					
Metrics	**SGM**	**GA-SF**	**GA-SC**	**GA-SCd**	**GA-US3D**	**GA-95SC**
$\text{Median}_{\text{diff}}$	1.40	0.94	0.33	0.28	0.61	**-0.05**
MAD_{diff}	3.27	1.92	1.45	1.27	1.10	**0.98**
3pix-acc(%)	57.0	62.3	69.3	72.3	78.3	**79.7**
1pix-acc(%)	32.3	28.9	36.9	38.7	36.3	**43.8**

$\text{Median}_{\text{diff}}$ and MAD_{diff} are defined in terms of pixels, while the accuracy is expressed as the percentage of all the pixels below the specified error threshold. Best result is indicated in bold font and underlined, second best just underlined.

accuracy, where the percentage of pixels whose difference with respect to the ground truth is below or equal to 3. Likewise, we estimate the 1 pixel accuracy. While a margin of 3 pixels for errors in the disparity map is acceptable for applications like autonomous driving, it would represent a large error when the depth is estimated from the aerial/satellite camera. For that reason, we also consider pertinent to analyze how this metric performs.

TABLE IV
RESULTS OF GANET FOR THE 4 K AERIAL DATASET

	Algorithms				
Metrics	**SGM**	**GA-SF**	**GA-SC**	**GA-SCd**	**GA-US3D**
$\text{Median}_{\text{diff}}$	-0.02	**-0.01**	-0.13	-0.12	0.56
MAD_{diff}	0.29	0.33	0.31	**0.28**	0.60
3pix-acc(%)	86.3	92.2	94.1	**94.3**	84.1
1pix-acc(%)	80.4	80.5	83.2	**84.0**	55.1

$\text{Median}_{\text{diff}}$ and MAD_{diff} are in defined terms of pixels, while the accuracy is expressed as the percentage of all the pixels below the specified error threshold. Best result is indicated in bold font and underlined, second best just underlined.

A. GANet Results

In Table III, we observe the results of using the GA-SF, GA-SC, GA-SCd, GA-US3D, and GA-95SC models for the US3D dataset. In addition, we also compared the results with the traditional SGM algorithm. It is important to mention that SGM does not produce a complete result, but has values only for those pixels where the estimation achieves the quality accepted by the algorithm. However, we evaluate the metrics in the whole image since completeness is a desired feature as well.

Considering the 3 pixel accuracy, we can observe that all the trained models outperform SGM by a significant margin. If we compare only GA-SF and GA-SC we notice already an improvement of 7% despite the shorter time that was used to train on the SyntCities dataset. GA-SCd has even more accurate results, but it also required a larger training and might not be a suitable option if the computational resources are limited. The model GA-US3D is even better by 6%, which is also expected since the domain gap does not play a role for this case. Interestingly, the GA-95SC model is the one that performed best, although it does not rely only on samples from the US3D dataset. While the improvement for the 3 pixel accuracy metric is slightly higher, the case for 1 pixel accuracy increases more than 7%. By comparing the results on the GA-95SC model and GA-US3D, the former had issues to estimate some areas, but produced sharper results than the latter. The training process augmented with the synthetic data seems to benefit from the accurate ground truth available on SyntCities. It is also important to remark that this strategy could work for datasets with reduced volume as well.

Focusing now on the 1 pixel accuracy, SGM has actually a better result than GA-SF but worse than GA-SC. In this way, we can notice how SC boosts accuracy to a finer detail. As mentioned before, this metric has special attention from the remote sensing community for a correct 3-D reconstruction. The values for $\text{Median}_{\text{diff}}$ and MAD_{diff} follow a similar trend to the accuracy.

Images to show the performance of the algorithms are presented in Fig. 4. The first row illustrates the disparity maps obtained and the respective reference. The second row shows the error maps, where all values ≥ 3 are in yellow. We can observe how completeness is obtained by all deep learning algorithms, which is not the case for SGM. However, the valid values obtained by SGM show good accuracy. Now, if we compare only the GA-SC and GA-SF cases for the disparity map, we can notice a better estimation for building areas and vegetation on GA-SC, as illustrated with the red rectangles. The model GA-95SC is of course the one with the best reconstruction, since it was partially trained on the test domain.

We can also study the performance for the error maps, where the presence of large areas in blue (error ≤ 1 pix) is desired. This is already achieved for many building and street sections on the GA-SC model as shown in the red rectangles. Difficult areas to solve for the model remain mostly for vegetation and vehicles, which in some cases were not present on the right view. In any case, the significant reduction of the error range would lead to a superior quality for DSM generation, crucial for remote sensing.

With regard to the results shown in Table IV for the 4 K aerial data we have a similar behavior. All models show a better accuracy for this dataset in comparison to US3D, this might be a result of the quality of the data referenced as a ground truth. Again, the neural networks outperform SGM, also for 1 pixel accuracy in this case. The GA-SCd model has a slight improvement with respect to the normal GA-SC. We did not compare the GA-95SC model because it would be challenging to evaluate the individual benefit of each of the two sources of the mixed dataset.

Nevertheless, we made inference on the GA-US3D model as this case is trained on real data as well. A 10% decrease in the 3 pixel accuracy of the result shows that training a model only on US3D data cannot be used for a different set of images, while the SF and SC datasets have a better generalization to estimate

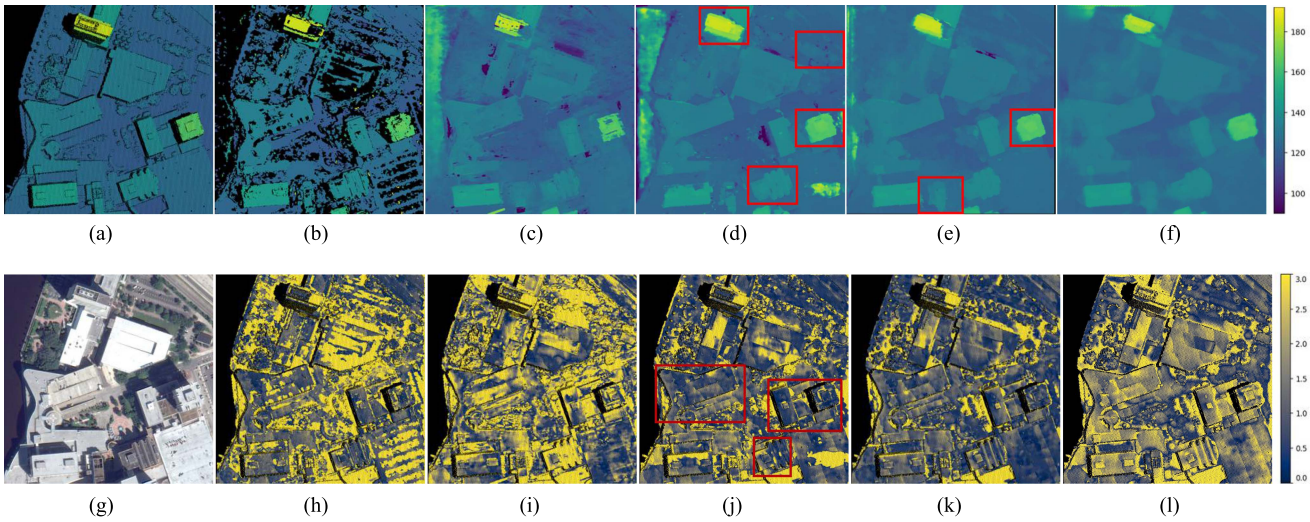

Fig. 4. Results from the GANet for the US3D dataset. The disparity reference (a) is compared to the disparity maps obtained by SGM (b) and the models GA-SF (d), GA-SC (d), GA-95SC (e) and GA-US3D (f). The range for the disparities is set from 90 to 192. Error maps for the reference RGB image (g) are shown for the same models SGM (h), GA-SF (i), GA-SC (j), GA-95SC (k) and GA-US3D (l). The error range is clipped to 0-3 pixels. (a) Disparity reference. (b) Disparity SGM. (c) Disparity GA-SF. (d) Disparity GA-SC. (e) Disparity GA-95SC. (f) Disparity GA-US3D. (g) RGB - Left view. (h) 3pix error SGM (i) 3pix error GA-SF. (j) 3pix error GA-SC. (k) 3pix error GA-95SC. (l) 3pix error GA-US3D.

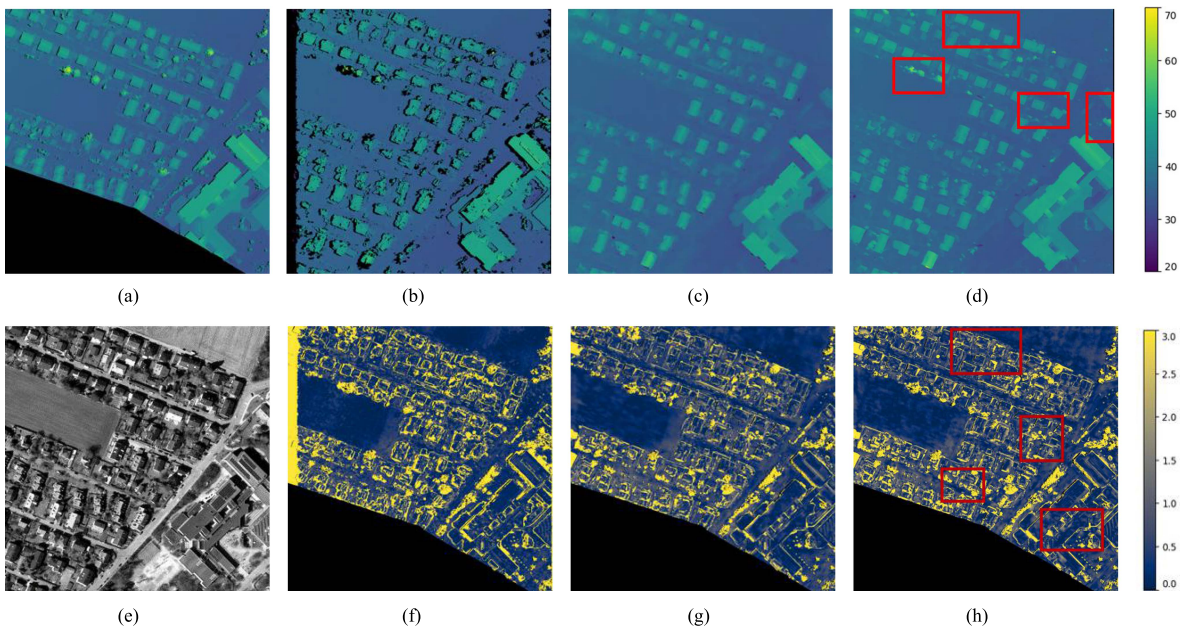

Fig. 5. Results from the GANet for the 4 K aerial dataset. The disparity reference (a) is compared to the disparity maps obtained by SGM (b) and the models GA-SF (d) and GA-SC (d). The range for the disparities is set from 20 to 70. Error maps for the reference optical image (e) are shown for the same models SGM (f), GA-SF (g) and GA-SC (h). The error range is clipped to 0-3 pixels. (a) Disparity reference. (b) Disparity SGM. (c) Disparity GA-SF. (d) Disparity GA-SC. (e) Optical - Left view. (f) 3pix error SGM. (g) 3pix error GA-SF. (h) 3pix error GA-SC.

disparities in different domains. Moreover, the accuracy in terms of 1 pixel is lower than any other case, including SGM that is not defined for all the pixels.

Visual results for the experiments on the 4 K aerial dataset are displayed in Fig. 5. Similarly to the US3D dataset, we notice more complete buildings and detection of vegetation on the GA-SC model. This is highlighted with the red rectangles. The effect is similar when analyzing the error map, where a significant part of the constructions is within 1 error accuracy and a larger number of trees is retrieved.

In all the illustrated cases, vegetation is still a challenging element in part because of seasonal changes, but we also think that a more realistic 3-D representation on the synthetic models could improve the performance.

TABLE V
RESULTS OF AANET FOR THE US3D DATASET

	Algorithms									
Metrics	**SGM**	**AA-SF**	**AA-SC**	**AA-US3D**	**AA-80SF**	**AA-80SC**	**AA-95SF**	**AA-95SC**	**AA-99SF**	**AA-99SC**
$\text{Median}_{\text{diff}}$	1.40	0.72	0.08	0.10	0.09	0.10	0.11	**-0.04**	0.09	-0.09
MAD_{diff}	3.27	1.82	1.72	**0.89**	1.08	1.05	1.25	1.09	1.42	1.23
3pix-acc(%)	57.0	63.2	64.3	**85.3**	78.5	79.6	74.8	77.7	70.9	74.7
1pix-acc(%)	32.3	29.3	32.6	**49.8**	41.7	42.4	37.5	41.4	34.6	38.4

$\text{Median}_{\text{diff}}$ and MAD_{diff} are defined in terms of pixels, while the accuracy is expressed as the percentage of all the pixels below the specified error threshold. Best result is indicated in bold font and underlined, second best just underlined.

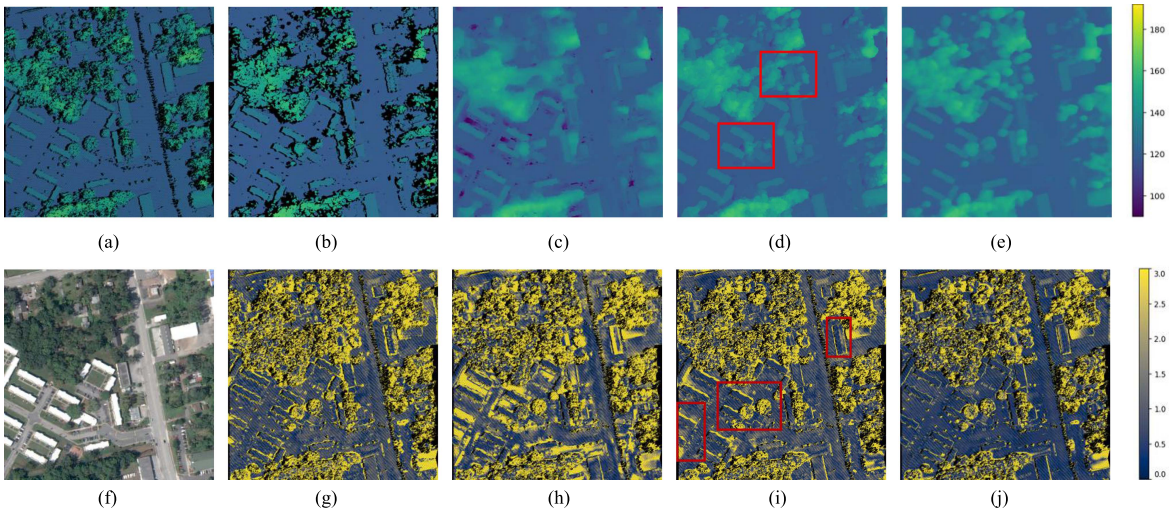

Fig. 6. Results from the AANet for the US3D dataset. The disparity reference (a) is compared to the disparity maps obtained by SGM (b) and the models AA-SF (d), AA-SC (d) and AA-US3D (e). The range for the disparities is set from 90 to 192. Error maps for the reference RGB image (f) are shown for the same models SGM (g), AA-SF (h), AA-SC (i) and AA-US3D (j). The error range is clipped to 0-3 pixels. (a) Disparity reference. (b) Disparity SGM. (c) Disparity AA-SF. (d) Disparity AA-SC. (e) Disparity AA-US3D. (f) RGB - Left view. (g) 3pix error SGM. (h) 3pix error AA-SF. (i) 3pix error AA-SC. (j) 3pix error AA-US3D.

B. AANet Results

Results from the implementation of the AANet architecture for the US3D dataset are shown in Table V. Accordingly, to the findings explained for the GANet, the deep learning models also outperform SGM. The highest accuracy is achieved by AA-US3D, which is an expected outcome taking into account that it is trained and tested on images of the same domain. Again, the AA-SC model got a better result than AA-SF and demonstrates the benefits of SyntCities for the training process.

There are also many cases presented with a mixture from the input data. Models with SceneFlow and SyntCities are compared at different rates of shared data. Nonetheless, the options where SyntCities is involved perform better than those with SceneFlow. This can be noted in both 3 and 1 pixel accuracy. Due to image size limitations not all the cases are illustrated.

Once more we appreciate the advantages of mixing the data with real samples. US3D has enough samples to be trained on its own imagery, but this might not be the case for other small datasets. Even by adding only 1% of real data to the training process we can reduce the domain gap, as exhibited in the last two columns of the table (comparing only with AA-SF and AA-SC).

Visual results related to these experiments are shown in Fig. 6. AA-SC generates sharper buildings and finer forest sections as remarked in the red rectangles. The range for disparities in the ground level is also more consistent with less generated discontinuities. Similar conclusions can be derived from the error maps displayed in the second row, where values for buildings and streets are more uniform on the AA-SC model. This is a congruous result for as much as the 3-D models were largely defined for these regions. Although there is room for improvement on the simulated urban scenes, the current quality of the synthetic samples suggests that its usage for training and pretraining is a feasible strategy. The vegetation is still a difficult area to address even for the AA-US3D model.

TABLE VI
RESULTS OF AANET FOR THE 4 K AERIAL DATASET

	Algorithms			
Metrics	**SGM**	**AA-SF**	**AA-SC**	**AA-US3D**
$\text{Median}_{\text{diff}}$	**-0.02**	-0.07	-0.06	0.29
MAD_{diff}	0.29	0.39	**0.28**	0.50
3pix-acc(%)	86.3	90.5	**92.7**	87.8
1pix-acc(%)	80.4	74.8	**82.6**	66.8

$\text{Median}_{\text{diff}}$ and MAD_{diff} are defined in terms of pixels, while the accuracy is expressed as the percentage of all the pixels below the specified error threshold. Best result is indicated in bold font and underlined, second best just underlined.

Turning to the results of the 4 K aerial view dataset shown in Table VI, the AA-SC model performs the best for both 1 and 3 pixels accuracy. An interesting point is the 1 pixel accuracy of SGM, which surpasses the one from AA-SF. This has also been

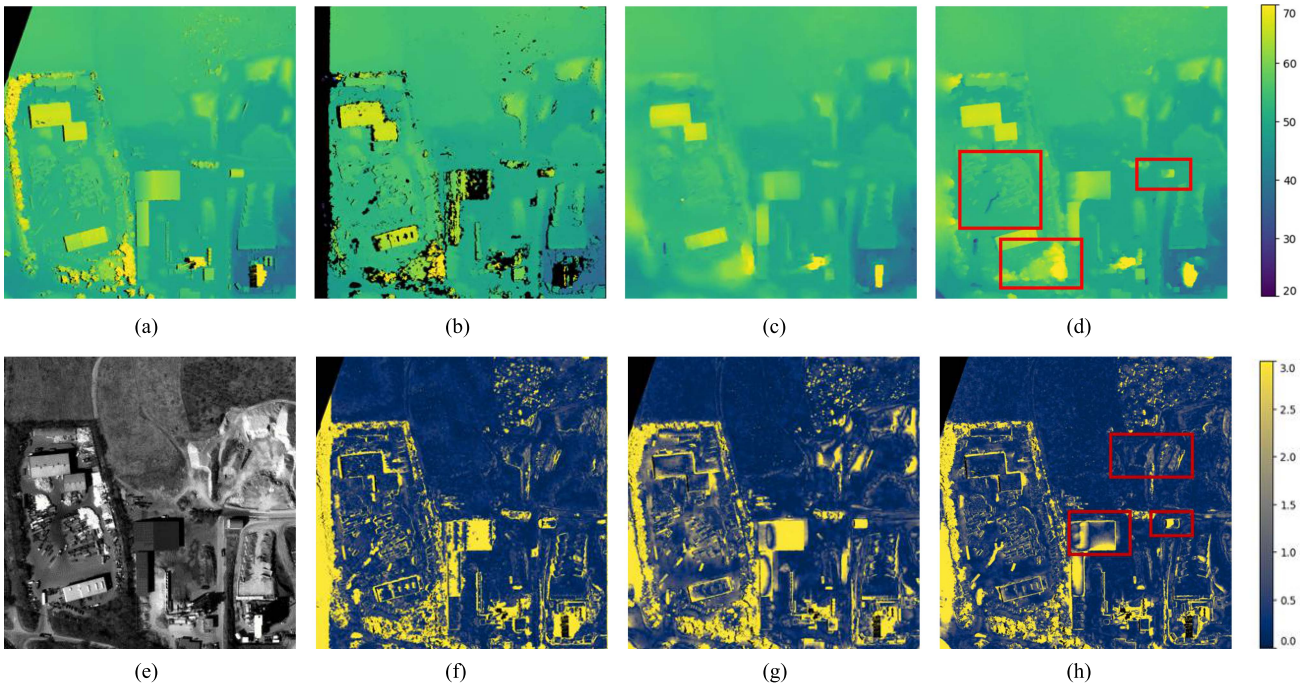

Fig. 7. Results from the AANet for the 4 K aerial dataset. The disparity reference (a) is compared to the disparity maps obtained by SGM (b) and the models AA-SF (d) and AA-SC (d). The range for the disparities is set from 20 to 70. Error maps for the reference optical image (e) are shown for the same models SGM (f), AA-SF (g) and AA-SC (h). The error range is clipped to 0-3 pixels. (a) Disparity reference. (b) Disparity SGM. (c) Disparity AA-SF. (d) Disparity AA-SC. (e) Optical - Left view. (f) 3pix error SGM. (g) 3pix error AA-SF. (h) 3pix error AA-SC.

observed in Tables III and IV. It seems that SyntCities raises the subpixel accuracy.

Images related to this experiment are on display in Fig. 7. In the selected sample vehicles are also present (see the largest red rectangle on the disparity maps) and finely estimated with the AA-SC model, where sharper boundaries are visible. AA-SC also has an improved representation for vegetation areas. The constructions have a similar performance to the other training experiments, exhibiting the benefits of the AA-SC models. Similarly to the results from GANet, the disparity maps generated on a model trained only on US3D data have larger errors than those trained on the synthetic data. Furthermore, the 1 pixel accuracy is again lower than the other compared methods.

An interesting point to mention for both GANet and AANet is the sensitivity to the disparity distribution of the training dataset. From the conducted experiments, we observed that the larger range covered by the synthetic datasets adapts easier for inference in unseen data. On the other hand, US3D has a narrower range and this would lead for a lower performance if the images are not preprocessed before inference on this model. We shifted the left image of the 4 K aerial samples to obtain a disparity distribution similar to the one of the US3D dataset to have a fair comparison. Without this preprocessing, a large systematic disparity offset has been observed. However, this behavior could cause worse results for other experiments if the data are directly fed into the networks without previous knowledge of the disparity distributions used in training. This will especially affect hilly or mountainous areas with larger disparity differences.

VI. Conclusion

A reliable DSM is a valuable resource for applications, such as city planning, updating of cadastral data, transport and flight simulation, autonomous driving or prevention and response to natural disasters, among others. Considering that, we presented in the current article the SyntCities dataset, which is to the best of our knowledge, the first large synthetic dataset for disparity estimation with focus on remote sensing. The generated samples include different illumination conditions and stereo configurations and benefit from the simulation model to generate a dense and accurate ground truth.

Experiments made for the disparity estimation demonstrate that the accuracy is improved by using our proposed dataset in comparison to models trained on the Scene Flow dataset. This was observed for both aerial and satellite data. A significant outcome is the boost for 1 pixel accuracy, which is desired for remote sensing applications where a single pixel might represent a large distance on the ground.

We also observed that our samples can be used as an augmentation strategy to compensate the lack of data in small real sets. Furthermore, models training on SyntCities without fine-tuning achieved a good performance on unseen data, such as the US3D and the 4 K aerial samples.

For future work, we want to upgrade the quality of the 3-D models by including not only urban areas but also features from natural landscapes, a more realistic vegetation representation and an expanded variety of buildings and architecture. We would also like to conduct some experiments to benefit from both disparity and semantic maps, since their information might be complementary. An algorithm able to create a labeled DSM would enhance many spatial databases.

Apart from that, the dataset could be enhanced with additional viewpoints to allow the training of multiview-stereo algorithms.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The authors would like to thank the Johns Hopkins University Applied Physics Laboratory and IARPA for providing the data used in this study, and the IEEE GRSS Image Analysis and Data Fusion Technical Committee for organizing the Data Fusion Contest.

The authors would also like to thank their colleagues Dr. F. Kurz and Y. Xia for providing and preprocessing the 4 K aerial samples and references, and C. Henry for his advice on the semantic segmentation parts.

References

[1] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, no. 1, pp. 7–42, 2002.

[2] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 328–341, Feb. 2008.

[3] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *J. Mach. Learn. Res.*, vol. 17, pp. 1–32, 2016.

[4] F. Zhang, V. Prisacariu, R. Yang, and P. H. Torr, "GA-Net: Guided aggregation net for end-to-end stereo matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 185–194.

[5] S. Wanner and B. Goldluecke, "Variational light field analysis for disparity estimation and super-resolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 606–619, Mar. 2014.

[6] N. Mayer et al., "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4040–4048.

[7] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 611–625.

[8] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.

[9] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3061–3070.

[10] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3234–3243.

[11] X. Li, K. Wang, Y. Tian, L. Yan, F. Deng, and F.-Y. Wang, "The ParallelEye dataset: A large collection of virtual images for traffic vision research," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2072–2084, Jun. 2019.

[12] M. Bosch, K. Foster, G. Christie, S. Wang, G. D. Hager, and M. Brown, "Semantic stereo for incidental satellite images," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2019, pp. 1524–1532.

[13] J. Liu and S. Ji, "A novel recurrent encoder-decoder structure for large-scale multi-view stereo reconstruction from an open aerial dataset," *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pp. 6049–6058, 2020.

[14] B. Le Saux, N. Yokoya, R. Haensch, and M. Brown, "2019 IEEE GRSS data fusion contest: Large-scale semantic 3D reconstruction [technical committees]," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 4, pp. 33–36, Dec. 2019.

[15] S. Kunwar et al., "Large-scale semantic 3-D reconstruction: Outcome of the 2019 IEEE GRSS data fusion contest—Part A," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 922–935, 2021.

[16] Y. Lian et al., "Large-scale semantic 3-D reconstruction: Outcome of the 2019 IEEE GRSS data fusion contest—Part B," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 1158–1170, 2021.

[17] F. Kong, B. Huang, K. Bradbury, and J. Malof, "The Synthinel-1 dataset: A collection of high resolution synthetic overhead imagery for building segmentation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2020, pp. 1814–1823.

[18] G. Yang, H. Zhao, J. Shi, Z. Deng, and J. Jia, "SegStereo: Exploiting semantic information for disparity estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 636–651.

[19] Y. Chen, W. Li, X. Chen, and L. Van Gool, "Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1841–1850.

[20] J. Zhang, K. A. Skinner, R. Vasudevan, and M. Johnson-Roberson, "DispSegNet: Leveraging semantics for end-to-end learning of disparity estimation from stereo imagery," *IEEE Robot. Automat. Lett.*, vol. 4, no. 2, pp. 1162–1169, Apr. 2019.

[21] P. L. Dovesi et al., "Real-time semantic stereo matching," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 10780–10787.

[22] Z. Wu, X. Wu, X. Zhang, S. Wang, and L. Ju, "Semantic stereo matching with pyramid cost volumes," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7483–7492.

[23] Q. Wang, D. Dai, L. Hoyer, L. Van Gool, and O. Fink, "Domain adaptive semantic segmentation with self-supervised depth estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 8495–8505.

[24] M. Denninger et al., "Blenderproc: Reducing the reality gap with photo-realistic rendering," in *Proc. Int. Conf. Robot.: Sci. Syst., RSS*, 2020.

[25] Q.-Y. Zhou, J. Park, and V. Koltun, "Open3D: A modern library for 3D data processing," *CoRR*, vol. abs/1801.09847, 2018.

[26] F. Kurz, D. Rosenbaum, O. Meynberg, G. Mattyus, and P. Reinartz, "Performance of a real-time sensor and processing system on a helicopter," *ISPRS - Int. Arch. Photogrammetry, Remote Sens. Spatial Inf. Sci.*, vol. XL-1, pp. 189–193, 2014.

[27] C. de Franchis, E. Meinhardt-Llopis, J. Michel, J.-M. Morel, and G. Facciolo, "An automatic and modular stereo pipeline for pushbroom images," *ISPRS Ann. Photogrammetry, Remote Sens. Spatial Inf. Sci.*, vol. II-3, pp. 49–56, 2014.

[28] A. Kendall et al., "End-to-end learning of geometry and context for deep stereo regression," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 66–75.

[29] J.-R. Chang and Y.-S. Chen, "Pyramid stereo matching network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5410–5418.

[30] F. Zhang, X. Qi, R. Yang, V. Prisacariu, B. Wah, and P. Torr, "Domain-invariant stereo matching networks," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 420–439.

[31] H. Xu and J. Zhang, "AANet: Adaptive aggregation network for efficient stereo matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 1959–1968.

[32] L. Lipson, Z. Teed, and J. Deng, "RAFT-Stereo: Multilevel recurrent field transforms for stereo matching," in *Proc. Int. Conf. 3D Vis.*, 2021, pp. 218–227.

[33] C. Wang et al., "Self-supervised multiscale adversarial regression network for stereo disparity estimation," *IEEE Trans. Cybern.*, vol. 51, no. 10, pp. 4770–4783, Oct. 2021.

[34] J. Dai et al., "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 764–773.

[35] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable ConvNets V2: More deformable, better results," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9308–9316.

[36] J. Höhle and M. Höhle, "Accuracy assessment of digital elevation models by means of robust statistical methods," *ISPRS J. Photogrammetry Remote Sens.*, vol. 64, no. 4, pp. 398–406, 2009.

Mario Fuentes Reyes received the bachelor's degree in mechatronics engineering from the National Polytechnic Institute of Mexico, Mexico City, Mexico, in 2016, and the master's degree in Earth oriented space science and technology from the Technical University of Munich, Munich, Germany, in 2018. He is currently working toward the Ph.D. degree in semantic 3D reconstruction from multi-view imagery with the Photogrammetry and Image Analysis Department, German Aerospace Center, Cologne, Germany.

His research interests include 3-D reconstruction, stereo matching, synthetic datasets, and deep learning techniques.

Pablo D'Angelo received the diploma (Dipl.-Ing FH) degree in computer engineering from the University of Applied Scienes, Ulm, Germany, in 2004, and the Ph.D (Dr.-Ing.) degree in computer science from Bielefeld University, Bielefeld, Germany, in 2007, with a dissertation on joint use of geometric and photometric methods for 3-D reconstruction from optical images.

From 2004 to 2007, he was with Daimler AG, Stuttgart, Germany, where he was involved in industrial computer vision. In 2007, he joined the Photogrammetry and Image Analysis Department, Remote Sensing Technology Institute, German Aerospace Center, Oberpfaffenhofen, Germany. His research interests include 3-D computer vision, photogrammetry and machine learning, and especially 3-D reconstruction from remotely sensed stereo imagery, with a focus on operational systems for large-scale image orientation and generation of digital elevation models.

Friedrich Fraundorfer (Member, IEEE) received the Ph.D. degree in computer science from the Graz University of Technology (TU Graz), Graz, Austria, in 2006.

He had a Postdoctoral stay with the University of Kentucky, Lexington, KY, USA, the University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, and ETH Zürich, Zürich, Switzerland. From 2012 to 2014, he was the Deputy Director of the Chair of Remote Sensing Technology, Technical University of Munich, Munich, Germany. He is currently an Associate Professor with the Institute of Computer Graphics and Vision, TU Graz, and associated with the Photogrammetry and Image Analysis Department, Remote Sensing Technology Institute, German Aerospace Center, Cologne, Germany. His research interests include 3-D computer vision, robot vision, and machine learning techniques.