remote sensing

MDPI

Article

Comparative Analysis of Deep Learning-Based Stereo Matching and Multi-View Stereo for Urban DSM Generation

Mario Fuentes Reyes [1,*], Pablo d'Angelo [1] and Friedrich Fraundorfer [1,2]

[1] Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), 82234 Wessling, Germany; pablo.angelo@dlr.de (P.d.); friedrich.fraundorfer@tugraz.at (F.F.)
[2] Institute of Computer Graphics and Vision, Graz University of Technology (TU-Graz), 8010 Graz, Austria; fraundorfer@icg.tugraz.at
* Correspondence: mario.fuentesreyes@dlr.de

Abstract: The creation of digital surface models (DSMs) from aerial and satellite imagery is often the starting point for different remote sensing applications. For this task, the two main used approaches are stereo matching and multi-view stereo (MVS). The former needs stereo-rectified pairs as inputs and the results are in the disparity domain. The latter works with images from various perspectives and produces a result in the depth domain. So far, both approaches have proven to be successful in producing accurate DSMs, especially in the deep learning area. Nonetheless, an assessment between the two is difficult due to the differences in the input data, the domain where the directly generated results are provided and the evaluation metrics. In this manuscript, we processed synthetic and real optical data to be compatible with the stereo and MVS algorithms. Such data is then applied to learning-based algorithms in both analyzed solutions. We focus on an experimental setting trying to establish a comparison between the algorithms as fair as possible. In particular, we looked at urban areas with high object densities and sharp boundaries, which pose challenges such as occlusions and depth discontinuities. Results show in general a good performance for all experiments, with specific differences in the reconstructed objects. We describe qualitatively and quantitatively the performance of the compared cases. Moreover, we consider an additional case to fuse the results into a DSM utilizing confidence estimation, showing a further improvement and opening up a possibility for further research.

Keywords: disparity estimation; depth estimation; urban reconstruction; digital surface models (DSMs); confidence estimation

Academic Editor: Haopeng Zhang

Received: 8 November 2024
Revised: 19 December 2024
Accepted: 21 December 2024
Published: 24 December 2024

Citation: Fuentes Reyes, M.; d'Angelo, P.; Fraundorfer, F. Comparative Analysis of Deep Learning-Based Stereo Matching and Multi-View Stereo for Urban DSM Generation. *Remote Sens.* **2025**, *17*, 1. https://doi.org/10.3390/rs17010001

1. Introduction

The task of generating DSMs is a first step in many remote sensing pipelines. Data from different sensors and platforms (usually aerial or satellite) can be used as input for this task, like images from traditional cameras, LiDAR or synthetic aperture radar (SAR). For this manuscript, we focused on the case where a DSM is created from optical imagery only, as this is often cheaper than the other sensors and offers sharp geometry for the reconstruction.

Currently deep learning based algorithms are state-of-the-art, however, many of these depend on supervised learning methods and a requirement for that is the availability of ground truth for training, which is still measured with LiDAR. This data acquisition is expensive and the quality of the ground truth depends on the density of the generated point cloud. Despite this issue, learning models have the advantage of being trained on a subset of data and tested on many other samples, so the ground truth is just required for the

training step, allowing the model to predict in many unseen samples. While the algorithms achieve in general a good reconstruction, their performance can be even improved by finetuning in some samples of the target dataset to reduce the domain gap (if any).

After obtaining a good dataset capable of training deep learning models, most existing network architectures are oriented towards either stereo matching or MVS approaches. While both are suitable for generating a DSM, they are based on different principles and therefore require different input data and network architectures.

The stereo algorithms require data that has undergone epipolar rectification, which means that the points to be matched are along the same epipolar line and we only consider candidates in one dimension. To calculate the height of objects in the scene, the baseline between the two images, the focal length of the camera, the position/orientation of the stereo array and the computed disparity map are needed.

MVS on the other hand does not need stereo rectified images, as it supports images from different points of view. Nonetheless, the correct relative position/orientation between the cameras is required for a homography warping. The algorithms estimate a depth map that can be converted into a height map based on the reference view position and rotation.

As deep learning architectures have evolved and achieved the best performance in the benchmarks, the differences between the two algorithms have become more pronounced. Datasets are designed separately for each case, as well as metrics and benchmarks. We already set the first experiments to evaluate both stereo and MVS algorithms in stereo paired images in our previous work [1], but we now explore multiple views and also test all the algorithms on real data. We use the available datasets SyntCities [2] and Dublin 2015 [3], where synthetic and LiDAR ground truth is available respectively. The aim was to make the comparison as fair as possible. This would highlight the differences between the algorithms. Metrics for all cases and discussions are presented for all the obtained results.

In the traditional pipelines for DSM generation, a set of candidate values is available for each pixel/location, which are later fused by using the median to determine a robust final value [4]. In practice, stereo methods are more widely used in remote sensing as they have been studied longer, just few pre-processing steps are needed and the matching works only along one dimension. MVS methods require less pre-processing steps and might benefit from the information provided by additional views, but they have been less studied. Deep learning algorithms are more robust in terms of matching, so MVS may achieve similar or better results than rectified stereo matching, despite its widespread use.

We explored beyond the traditional fusion, by using a confidence estimation which could help to pre-select the best candidate values before fusion. The confidence estimation responds to one of the remaining issues of deep learning, the fact that there is a prediction for each pixel, whether this is a reliable one or not. The confidence estimation aims to give a value related to this certainty, which we use to sort the available height values used to be fused in the DSM. Although the improvement in the DSM accuracy is small, the experiments show that there is potential for further research in this direction.

Summarizing, our main contributions are:

- A fair comparison of learning-based stereo and MVS methods while using multiple views/stereo-pairs for the same region.
- We evaluate the algorithms in synthetic data, where the ground truth is highly accurate and on the real images, as an application case with challenging regions.
- We explore an alternative way to fuse the height values into a DSM by using the confidence associated to each prediction made by the neural networks.
- We share the processed Dublin dataset [3] to have a large dataset compatible with stereo and MVS algorithms (The processed Dublin dataset can be downloaded at: https://zenodo.org/records/12772927, accessed on 20 December 2024).

2. Related Work

In this part we describe some of the main algorithms and neural networks applied to the tasks of stereo matching and MVS highlighting their differences. Besides, we introduce some available algorithms for the confidence estimation in the stereo matching case.

2.1. Stereo Methods

Prior to deep learning solutions, stereo algorithms were mostly based on a cost volume generation pipeline and its refinement to produce smooth results. Usually the steps for stereo estimation are matching cost computation, cost aggregation, disparity estimation and disparity refinement [5]. A widely used algorithm for stereo matching is Semi-Global Matching (SGM) [6], which can be implemented also to work in real-time due to its compromise between efficiency and accuracy. As it is the case with non-learning algorithms, it can be applied to any pair of images without prior knowledge and produce a good quality result. Nonetheless, the tuning of the penalty parameters has a strong influence on the performance of the algorithm.

Recently, deep learning solutions have been the leading approaches for stereo matching. MC-CNN [7] replaced the matching cost computation of the traditional pipelines with a neural network and refined the computed cost volume with SGM to reduce the impact of the remaining outliers, showing a good performance especially in terms of smoothness for the computed disparity map. Later on, end-to-end networks were developed to predict the disparity maps from the stereo images, learning also the refinement steps. The first approaches were DispNet [8] with an encoder-decoder architecture and GC-Net [9] that incorporated 3D convolutions. Among the architectures that are widely known and used as a baseline to compare performance, we can mention GANet [10], AANet [11] and DSMNet [12]. GANet is a learning-based implementation similar to SGM, where the penalty parameters are learned and 3D convolutions are used to refine thin structures. AANet produces smooth results and avoids the expensive 3D convolutions using less memory than GANet with a slight loss in accuracy. DSMNet on the other hand, tried to reduce the domain gap by using a domain normalization.

Newer architectures benefit from more complex architectures. RAFT-Stereo [13] adds gated recurrent units (GRUs) for a robust result in difficult areas, like textureless sections. Besides, it is less affected by the domain gap problem. A different strategy is STTR [14], where transformers are included and the network also alleviates the constraint of a fixed disparity range. Unimatch [15] proposes a unified model able to address optical flow, stereo matching and depth estimation. This network is based on transformers for feature similarities instead of convolutional layers. EPNet [16] focuses on recovering small and thin structures present in the images by using an additional encoder for edge preservation and a coarse-to-fine strategy for the depth estimation. Selective-Stereo [17] introduces an architecture including Selective Recurrent Units (SRUs) to recover finer details and capture low-frequency information in smooth regions.

In our manuscript, we will use only AANet as this requires less time for training/inference than other networks, produces a good quality result, and is a common baseline to compare new methods.

2.2. MVS Methods

The multi-view networks do not require the input images to be on the same epipolar line and therefore allow the reconstruction to be based on multiple points of view. Such a reconstruction takes place directly in the 3D space, so the predictions represent the distance from the camera plane to the objects as in the traditional sweep plane algorithms. In contrast to stereo methods, the MVS approaches require a estimated depth range as well

as the relative camera positions and rotations values. MVS algorithms can use two or more views, so it is important to specify how many of these are being used when implementing the algorithm.

Non-learnable photogrammetric algorithms have been also developed for this task. COLMAP [18] reconstruction benefits from multi-view geometric consistency, and its algorithm to sort the additional views (with respect to a reference view) is used also by deep learning solutions as a starting point. GIPUMA [19] applies an iterative process in the 3D space which is computed efficiently by using GPU resources.

Deep learning architectures have also been leading the MVS benchmarks in the last years, especially in terms of completeness. MVSNet [20] is a pioneering work that implements the plane sweep algorithm in a learnable way. R-MVSNet [21] includes GRUs which help to slightly improve the results. Another strategy is CasMVSNet [22], that follows a coarse-to-fine architecture reducing the memory consumption and allowing higher image resolutions. VisMVSNet [23] incorporates information related to the occluded pixels to rely in visible pixels for a more robust reconstruction. RA-MVSNet [24] focuses on textureless areas and complex boundaries by using both the depth and signed distance field (SFD) in the cost volume. CL-MVSNet [25] adds two parallel branches in the network. The first one is image-level and aims for better context awareness and the second is scene-level for robustness regarding view-conditional differences. GeoMVSNet [26] includes geometrical information from fine and coarse stages for a more robust prediction. It also applies a frequency domain filter in the depth maps at different stages. GC-MVSNet [27] also highlights the benefits of using geometrical information by adding a geometrical consistency loss. UniMVSNet [28] has a depth representation that allows the network to consider both a classification and a regression task simultaneously, leading to significant improvements in the performance. On top of that, computational resources are less demanding than for other networks. Therefore, we select UniMVSNet for the experiments in this manuscript.

2.3. Confidence Estimation

The confidence estimation is a research area that has been explored already in the task of stereo matching. Given a disparity map, which is predicted with a neural network (or a photogrammetric algorithm), the confidence estimation aims to give a value that is related to the certainty of the prediction for each pixel in the result. This would be similar to some post-processing steps applied in the stereo matching, such as left-right check consistency, where according to the bilateral reprojection of the images using the disparity maps, some disparity predictions are discarded due to inconsistencies.

As with the disparity and depth estimation tasks, the confidence can also be estimated by learnable and non-learnable algorithms. Regarding the latter ones, one of the first quantitative evaluations is shown in [29]. Most of the evaluated algorithms are based on the cost volume used to estimate the disparity values. Confidence for each pixel can be computed directly from the cost, by evaluating the curvature of the cost curve, analyzing the presence and distribution of the local minima, the behavior of the whole cost curve or by using the left-right consistency as already mentioned.

With respect to learned-based algorithms, a quantitative evaluation can be found in [30]. These algorithms take as input the input reference image, the predicted disparity maps and/or the cost volume, although the latter increases significantly the memory consumption in the implementations. CCNN [31] was one of the first architectures designed to predicted confidence maps by using Convolutional Neural Networks (CNNs) and Fully Connected Networks (FCNs). Since this method did not use the cost volume as input, it is more flexible to test in other stereo matching algorithms. PBCP [32] used a patch based solution on maps predicted by SGM and significantly reduced the confidence

prediction error. PKRN+ [33] included layers able to capture not only the information for the computed pixel, but local context to estimate the confidence. In this way, regions with similar confidence values are smoother. A different architecture [34] proposed to use not only the disparity map, but the cost volume as input for the network. To reduce the high computational cost of processing the entire cost volume, this volume is inverted (to represent similarities instead) and only the highest matching candidates are selected using the "top-k" operation from PyTorch. Finally, LAFNet [35] takes reference image, disparity map and cost volume (with the same preprocessing as [34]) as inputs and includes convolutional spatial transformers in the architecture, leading to a remarkable performance between the state of the art solutions. Hence. we selected LAFNet for our experiments related to the confidence-based estimation.

Since LAFNet requires the cost volume as input, we had to select a neural networks that are based on a cost volume approach. The previously selected networks for disparity and depth estimation, namely AANet and UniMVSNet were also chosen because their cost volumes can be exported to be used as input for LAFNet. Although LAFNet has been designed exclusively for disparity maps and no for the MVS case, we explored using the depth maps with their respective cost volumes as input a in a similar manner to stereo data.

3. Datasets

As mentioned in the introduction, datasets for stereo and MVS algorithms have been designed separately for each task, making it difficult to establish a common dataset to assess the performance reconstruction of both approaches. To overcome this obstacle, we decided to prepare two datasets for our experiments. First, we used SyntCities as in our previous work [1], but instead of using only two views for all cases, we selected additional views and different baselines. Second, we also evaluated the performance of the algorithms on real data, so we processed the Dublin dataset [3] to be compatible with both approaches and generated the required ground truth. Detailed information is given in the next sections. We focused on aerial data as the resolution and quality of the ground truth help to evaluate the ability to reconstruct finer details like small objects and sharp edges.

3.1. SyntCities Dataset

The SyntCities dataset is a synthetic dataset that was developed to compensate for the lack of stereo paired data in the remote sensing field. Since these images are generated directly from the 3D software Blender (v3.1) by using BlenderProc [36], the ground truth is accurate and dense, which means we have a reliable reference value for all pixels. The images have been rendered at a ground sample distance (GSD) of 10 cm, 30 cm and 1 m. In the original setting, 4 pairs are given for the same area with different baselines. For the new experiments, we benefit from the fact that despite having different baselines, all tiles with the same naming number (based on the SyntCities file organization) are on the same epipolar line. The SyntCities dataset assumes that the camera follows a flight track over the scene and acquires the images at 25 locations; as those points act as the center for the stereo arrays, we generated the stereo pairs by simply increasing the baselines. Hence, for each location we have 8 images along the epipolar line considering the left and right views (4 baselines $\times$ 2 views). The selected testing samples have a GSD of 30 cm and 1 m and belong to the Venice and Paris samples, as height differences are not so large in these cities.

In our experiments, we used a maximum of 6 views for each location. Due to the camera parameters of the stereo pairs, all images cover approximately the same area on the ground, as shown in the Figure 1, where all the cameras are pointing to a common area. Assuming that we select V_N ($N \in [1, 6]$) as the reference view, we have 5 additional views to help for the reconstruction of V_N. The distance between the cameras is given in

the image as baselines with respect to V_1. The cameras were not rotated nor displaced out of the epipolar line. As SyntCities included ground truth only for the default stereo pairs, we generated the missing disparity maps from the depth maps (available for all views) and the camera parameters. Apart from that, no additional data is required.

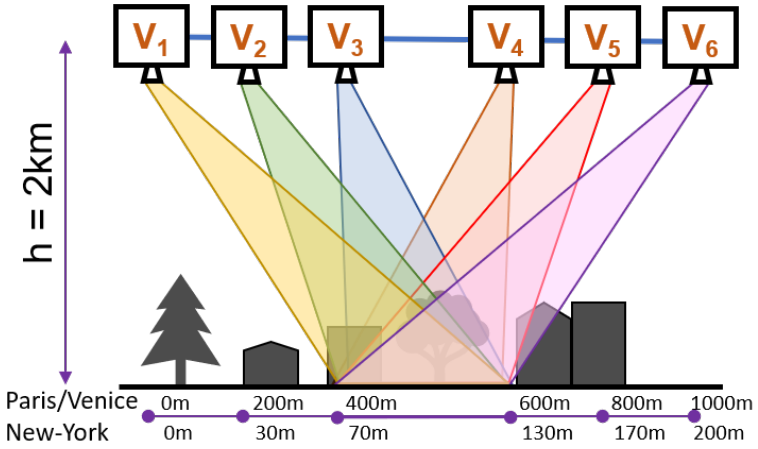


Figure 1. Selected geometry for SyntCities samples. All images lie on the same epipolar line with different baselines. There are 6 available views for each region on the surface. Baseline distances are given with respect to V_1.

3.2. Dublin Dataset

The Dublin dataset (The original Dublin dataset can be downloaded at: https://geo.nyu.edu/?f%5Bdct_isPartOf_sm%5D%5B%5D=2015+Dublin+LiDAR, accessed on 20 December 2024) is a collection of data acquired on 2015 over the downtown of Dublin, Ireland. The campaign had a flying altitude of 300 m and retrieved LiDAR data (as point clouds and full waveform), oblique images, geo-referenced RGB and infrared imagery, and the respective acquisition metadata.

As a first step, we downloaded all the point clouds and merged them to create a single DSM, as the ground truth was later computed from it. The DSM was created with a GSD of 10cm and is shown in Figure 2. Due to the sensor acquisition not all the pixels will have a ground truth, but for those where the value is defined, this is computed from a dense measurement, offering a good quality ground truth. Since the reference DSM is calculated from the original LiDAR point clouds, moving objects such as cranes may be measured in more than one location. However, the density of such objects in the dataset is low.

We selected the georeferenced RGB imagery as input for our experiments. The original images had a size of 9000×6732 pixels with a GSD of 3.4 cm. We downsampled the images by $\times 9$, changing the images to a size of 1000×748 pixels with a GSD of 30.6 cm, similar to the one in SyntCities. With the downsampled size, it is also easier to use the images as input for the neural networks without additionally cropping and merging the tiles for pre and post processing.

The data was further processed for the two input cases: Dublin_stereo and Dublin_MVS. A diagram for the applied pipeline is shown in Figure 3, where we have K input images. In the case of the Dublin_stereo dataset, we selected a pair N of the K downsampled images, the pair had to be epipolarly rectified for stereo matching. For each image, we selected the 5 closest acquisitions (based on the Euclidean distance of the positions) to set the pairs. The epipolar rectification is done with the compact implementation described in [37]. Once the pair has been rectified, we use a photogrammetric algorithm to convert from the DSM to a disparity map, which is aligned to match the "left" image of the pair (so the disparities have a positive range as required for the networks). Occlusions are handled by utilizing a DSM with higher resolution than the images and keeping only points closest

to the image. Hence, the stereo dataset includes pairs of rectified images with the respective disparity ground truth. Two example data pairs of the Dublin_stereo dataset are shown in Figure 4.

Figure 2. Dublin digital surface model obtained by merging all provided point clouds and used as ground truth. Blue areas are low objects and red areas are high objects.

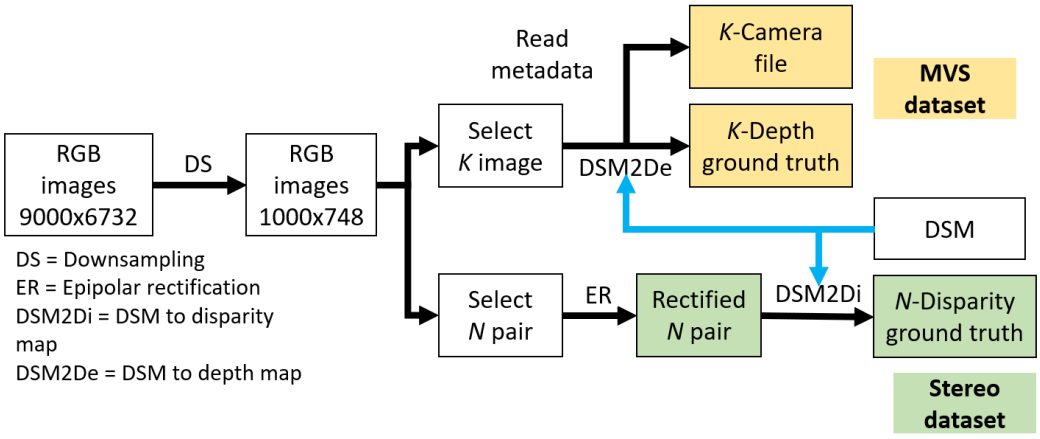
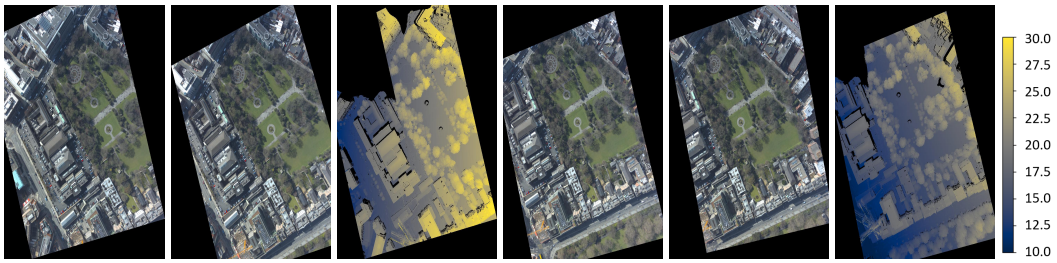


Figure 3. Pipeline used to generate the Dublin dataset for both cases: Dublin_stereo and Dublin_MVS.

With respect to the Dublin_MVS dataset, after downsampling the images, we processed the camera values for positions and rotations from the metadata to be compatible with the format required for the camera files in the MVS approaches, which includes camera extrinsics, intrinsics and an estimated depth range where the scene is located. The depth range is computed from the DSM, with a range that includes $\mu \pm 4\sigma$, being μ and σ the mean and standard deviation of the depth values according to the camera parameters. This range is different for each image. Note that the tiles used here have not been epipolarly rectified (unlike Dublin_Stereo) and correspond to the original points of view.



(**a**) Left A (**b**) Right A (**c**) GT A (**d**) Left B (**e**) Right B (**f**) GT B

Figure 4. Dublin_stereo dataset samples. (**a**,**d**) are the left views for the corresponding (**b**,**e**) right views, (**c**,**f**) are the ground truth aligned with the left views. Bar scale for disparities is in pixels.

The depth ground truth is obtained in a similar way to the stereo case, where we used the DSM and photogrammetric relations to convert the DSM into the depth map for each image. As the depth map does not depend in the additional views, it is always the same for a specific image and we do not need to provide ground truth for different image pairing. Therefore, the MVS dataset includes the RGB images with the respective depth map and camera file. An example of the images included in this dataset are shown in the Figure 5.

(**a**) Image A (**b**) Depth GT A (**c**) Image B (**d**) Depth GT B

Figure 5. Dublin_MVS dataset samples. (**a**,**c**) are the reference views for the corresponding (**b**,**d**) ground truth. Bar scale for depth is in meters.

The Dublin dataset acquisition track has a different geometry to the one presented for SyntCities. For the Dublin campaign, images are taking with a single camera along the flight path. Therefore, the images cover different areas with some overlapping between adjacent acquisitions. In the Figure 6 we show a simplified diagram of the camera positions and ground coverage. A distance of approximately 100 m is given between two consecutive images, leading to a forward overlap of ∼70%.

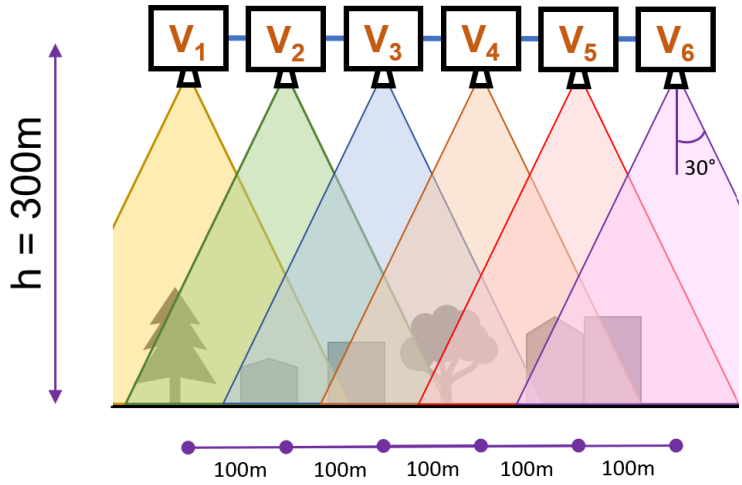


Figure 6. Selected geometry for Dublin samples. Images lay on a flight path with an approximate baseline of 100 m, but not in the same epipolar line.

Unlike the SyntCities case, in the Dublin dataset some regions are not visible in adjacent input views, which makes the matching more challenging than for the synthetic data. Moreover, the density of objects and textures in the Dublin dataset is larger, posing additional difficulties for the reconstruction algorithms.

4. Methodology

In the following paragraphs we describe the process used to fuse the data (with and without confidence guidance), as well as the training conditions of the applied stereo and MVS networks. For the MVS network, we considered two cases, applying it as a stereo matching algorithm (which means many input stereo pairs) and as a full multi-view algorithm (where many views are taken simultaneously as input). Hence, we analyzed three cases, namely: Stereo, MVS_Stereo and MVS_Full. For a clear explanation of the difference between the last two, please see Section 4.4.

It is relevant to explain the reasons why we specifically selected AANet and UniMVS-Net for our experiments. We already mentioned some arguments, namely short inference time, memory efficiency, the cost volume based architecture and the advantage that these are usually baselines to compare newer architectures. It is difficult to select from all existing architectures a set of them that can be easily compared. However, these two networks share the following elements:

- The initial layers of each network create feature volumes relevant for the matching.
- The cost volume is designed to have a single channel per disparity/depth candidate value, unlike other architectures where multiple channels represent each candidate. This is a critical aspect, as the cost volumes used as input to the confidence networks require the single channel shape. Newer approaches based on GRUs or Transformers might present a compatibility issue.
- The architecture follows a coarse-to-fine design which is also memory efficient.
- The predicted disparity/depth maps are generated at full resolution, without the need for further upsampling algorithms.
- The design of the networks is based on traditional convolutions.
- Although adapted for a learning scheme, the working principle is based on conventional stereo and MVS approaches, such as SGM and the plane sweep algorithms.

We did not include more architectures for each case, as it is out of the scope of this article to evaluate the performance of multiple stereo and MVS approaches, but to observe the main differences between these two. In addition, these two networks were compared with traditional approaches in our previous work [1], which complements the findings from the experiments in this article.

4.1. Predicted Maps Fusion

Different methods can be used to estimate the disparity/depth maps as a first step to generate a DSM. However, due to memory and computational constraints, remote sensing images are usually cropped into tiles, which may correspond to different regions with some overlapping. Hence, the predicted results define a stack of smaller DSMs that need to be aligned and fused into a single DSM. To achieve this fusion, steps are different for stereo and multi-view cases.

The pipeline to fuse predicted disparity and depth maps is shown in the Figure 7. We represent here a case to fuse 6 images of SyntCities, but the principle is the same for the Dublin data.

Starting from the stereo cases, which are Stereo and MVS_Stereo, we have a total of 15 possible combinations, and we always consider the disparity map from left to right to get positive values, which is a restriction for the estimation of the networks. The

15 disparity maps are then converted into height using the camera parameters along with the baseline and subsequently georeferenced using the camera positions. Nonetheless, the transformation of the disparity maps to height maps is still influenced by the acquisition perspective, having an oblique view. Hence, it is necessary to orthorectify the images to have the geometry required for the DSM.

We also have the MVS_Full case. Using the algorithms for MVS estimates the depth for only one of the views at a time, which is considered to be the reference view while the additional views provide complementary information. This means that we obtain 6 depth maps as a result of giving the same number of input images, since each of these 6 input images is used once as the reference view with the remaining ones used as the complementary views. Although the number of results may seem smaller than in the stereo case, the same number of images is used within the algorithms. After estimating the depth for each view, we transformed this into height using also the camera parameters. Similarly to the stereo case, the height map is still oriented to match the camera perspective and required orthorectification as well.

Having all the results as orthorectified height maps, it is now possible to fuse the results into a single DSM, benefiting from all single estimations. We considered two basic yet widely used methods: mean and median for each pixel/location. The former provides insights of the distribution of the predicted results. The latter is more effective and makes a robust fusion by avoiding the influence of outliers, being the most common strategy.

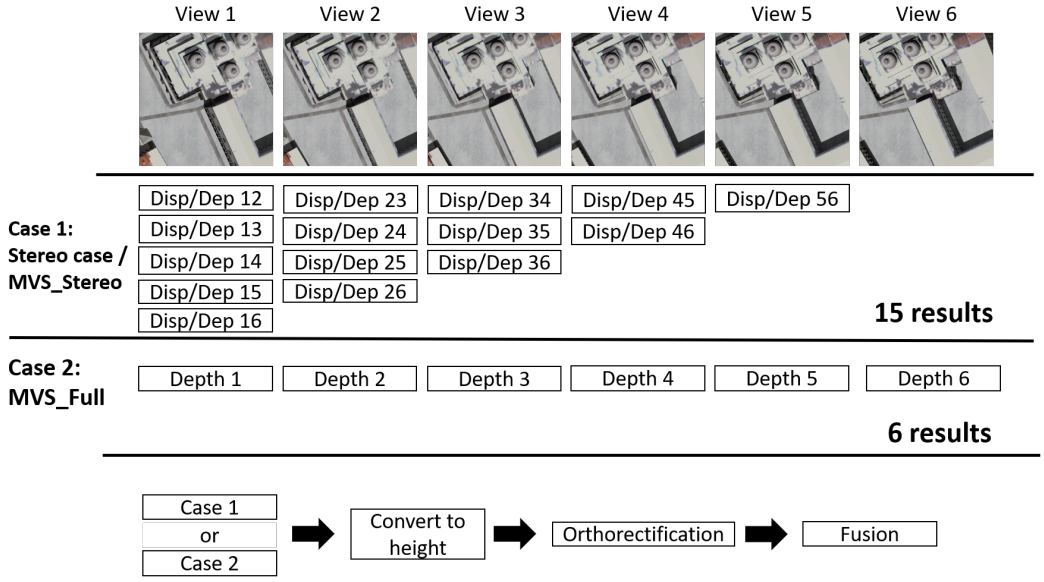


Figure 7. Pipeline used to fuse the results of the predicted disparity/depth maps. In the case of the Stereo and MVS_Stereo methods, more results are available but they use the same available information as the MVS_Full case. All results then follow the same steps which include height conversion, orthorectification and fusion.

4.2. Confidence Based Fusion

We also analyzed the case of fusing the depth and disparity maps using a confidence based fusion. A diagram to explain the process is shown in the Figure 8, but we describe here the steps in detail. The confidence maps help to fuse the depth and disparity maps, so we need to process all the data simultaneously.

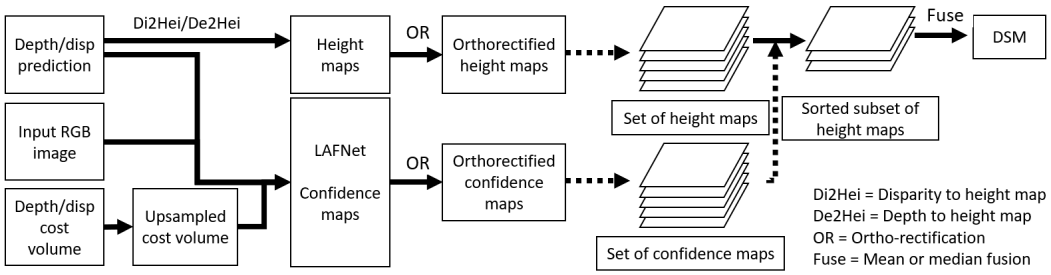


Figure 8. Pipeline for confidence-based fusion. After estimating confidence maps along with the height maps obtained from the reconstruction algorithms, a stack of height maps is sorted based on the respective confidence values and then we compute the median to get the final DSM.

First, disparity/depth maps are converted to height maps using photogrammetric algorithms. For this step, LAFNet is not required, just the results from the Stereo, MVS_Stereo and MVS_Full algorithms. In parallel, the same depth/disparity maps along with the cost volume (which has to be upsampled) and the RGB images are used as input to LAFNet, generating a confidence map as a result.

After that, both height and confidence maps are orthorectified. Since both maps are obtained for the same regions, the orthorectified maps cover the same pixels/areas. If we apply these two steps to all input depth/disparity maps, we end up with a stack of height and confidence maps.

In the above fused cases, we only apply the median to all the candidate height values for each pixel to obtain the fused height. We do propose a different strategy to fuse the height values by using the corresponding confidence values. We sort the stack of confidence maps according to the values for each pixel from higher to lower, and based on this sorting, we re-arrange the stack of height values as well. Afterwards, we remove the less confident height values according to a removal percentage ($rem_{\%}$). For example, if we have 10 height values for a certain pixel and set $rem_{\%} = 50$, only the 5 candidates with higher confidence remain. We compute the median from the remaining values to generate the DSM.

4.3. Stereo Training

We train AANet for stereo matching in both SyntCities and Dublin (stereo dataset), training from scratch for SyntCities and used this model to finetune on the Dublin data. We followed this strategy as the ground truth for SyntCities is dense and accurate, so the finetuning would help to reduce the domain gap for the testing area. For SyntCities, from the original 5400 images from the training subsets, we removed 300 cases with large baselines, keeping 5150 for training. 22 samples from the test subsets with 5 views each, so 110 samples were used for testing. The 5 additional views are on the same epipolar line, so they can be used in stereo or multi-view mode. These images are taken from 3 stereo pairs (6 images in total) where the leftmost view is used as reference. In the case of Dublin, from the available tracks, we selected the subset 150326_122941 for finetuning and the subset 150326_120403 for testing.

The training for SyntCities takes different views along the epipolar line as explained previously for Figure 1. We used a batch size of 20 and trained the model for 370 epochs and call this model Stereo_SC. The finetuning is done with the Dublin stereo samples for additional 500 epochs. We reduce the maximum disparity to 96 as this range is enough for these samples. We call this model Stereo_Du. Training was conducted on 4 × NVIDIA GeForce RTX 2080 Ti GPUs.

4.4. MVS_Stereo and MVS_Full Training

Similarly to AANet, we train firstly on SyntCities and then finetuned the model on Dublin samples. However, we apply two different training models for UniMVSNet: as a stereo matching case and full multi-view, which means 2 views and 6 views as inputs respectively. The first case will help to study the performance of UniMVSNet with conditions very similar to AANet, and we call this case MVS_Stereo. The full multi view is intended to give data to compare the impact of having more views as input and if this is beneficial for the reconstruction and we named this case simply MVS_full.

In the MVS_Stereo based instance, we train UniMVSNet on SyntCities for 40 epochs with 2 input views, a batch size of 2 and the image pairs are loaded with the same pairing order as for AANet. Afterwards, we finetuned the model for additional 270 epochs. We call these models MVS_Stereo_SC and MVS_Stereo_Du for SyntCities and Dublin respectively.

Similarly, we train the MVS_Full case with UniMVSNet by applying a number of views of 6 for 160 epochs. The number of iterations is larger as there are less possible combinations of input images as for the stereo case. For the finetuning we applied additional 600 epochs. These models are named as MVS_Full_SC and MVS_Full_Du. Finetuning models had more epochs due to the relatively fewer samples in Dublin comparing to SyntCities.

4.5. LAFNet Training

LAFNet requires the cost volumes as inputs along with the RGB images, the predicted depth/disparity maps and the depth/disparity ground truth maps. While using algorithms such as SGM or MC-CNN, the whole cost volumes are easy to identify and export as additional files, providing also information for each pixel. However, neural networks usually use structures where the volumes are downsampled to reduce computational resources. Moreover, the volumes in the coarsest resolutions generally offer a better overview of the matching, as they take into account the full disparity range. The finer volumes mostly refine around a certain disparity range, not the full one. Hence, we used the coarsest cost volumes from AANet and UniMVSNet, in both cases after the aggregation steps to reduce the presence of outliers.

We adapted both networks to export the cost volumes as described above. Besides, LAFNet applies a pre-processing step to the input cost volumes as mentioned in [34], where the values are normalized to improve the discriminative power of the network and the "top-k" function selects the main cost candidates only. This helps also to reduce the memory demands of the algorithm. In order to also reduce the storage space required for the cost volumes, we apply this processing step before exporting the cost volumes. It also avoids additional processing each time the LAFNet is loading the data.

Nonetheless, using the coarse cost volume makes the input data to be mismatched in terms of resolution. We solved this by interpolating the stored coarse cost volume to match the input image. A more sophisticated upsample strategy based on learning parameters might provide a better result, but we keep that out of scope as our purpose is not to design a new confidence learning network.

We also observed that LAFNet uses a binary cross entropy loss to segment the confidence mask into the ideal case of confident and non-confident pixels. Still, we would like to study the effect of using L1-loss based on the error instead. The confidence estimation is based on an error threshold (common values for disparity threshold errors are 3 and 1 pixels) and is computed from the difference between the predicted and ground truth disparities as:

$$diff = \begin{cases} |disp - disp_{gt}| & \text{if} \quad |disp - disp_{gt}| \leq err_t \\ err_t & \text{if} \quad |disp - disp_{gt}| > err_t \end{cases} \tag{1}$$

$$conf = 1 - \frac{diff}{err_t} \tag{2}$$

where err_t is the error threshold, $disp$ the predicted disparity value, $disp_{gt}$ the ground-truth disparity value and $conf$ the confidence value used as ground truth for LAFNet. Due to the clipping of the disparity difference ($diff$), the values of confidence are restricted to $0 \leq conf \leq 1$.

Since the real data is more challenging and the confidence can help to distinguish bad predicted areas, we trained only on the Dublin dataset. We trained LAFNet for 250 epochs, with patches of 494×494 pixels and a batch size of 4. Tiles are cropped from all the inputs over the same pixels to maintain consistency with the ground truth. Such tiling is applied due to memory constraints. The LAFNet models were trained on one NVIDIA GeForce RTX 2080 Ti GPU and we call this model Conf_Stereo. The original input cost volumes, which were obtained with AANet, were upsampled by $\times 3$ to match the images input size. For the results coming from UniMVSNet, we upsampled $\times 4$ the input cost volumes, and these models were trained for 350 and 1000 epochs for the MVS_Stereo and MVS_Full cases respectively, naming them as Conf_MVS_Stereo and Conf_MVS_Full. The latter had more epochs as the number of input depth maps is lower than the former.

5. Results

In this section we present the qualitative and quantitative evaluation of the fused models in comparison to the ground truth DSM. For the three applied algorithms (Stereo, MVS_Stereo and MVS_Full) we used both datasets SyntCities and Dublin, having a total of 6 DSMs to be evaluated.

5.1. Metrics

We consider three metrics to evaluate the accuracy of the fused models, which are:

- Median Absolute Deviation (MAD). Since the median based metrics are more robust to outliers [38] we apply MAD, which can be derived from the median of the difference (Med_{diff}). The median of the difference is computed between the ground truth and the fused DSMs. This is computed as:

$$\text{Med}_{\text{diff}} = \text{median}(X_{\text{diff}}), \quad X_{\text{diff}} = X - \bar{X} \tag{3}$$

  where X is the ground truth, $\bar{X}$ is the compared DSM and X_{diff} is the difference between both. Second we compute the MAD as:

$$\text{MAD}_{\text{diff}} = \text{median}(\mid X_{\text{diff}} - \widetilde{X}_{\text{diff}} \mid) \tag{4}$$

  where $\widetilde{X}_{\text{diff}} = \text{median}(X_{\text{diff}})$
- Mean Absolute Error (MAE), which is the absolute difference between the predicted result and the ground truth values. It is computed as:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |X_i - \bar{X}_i| \tag{5}$$

- Root mean square error (RMSE). It helps to remark the presence of large outliers, as they get more weight in the metric. This can be computed as:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X}_i)^2} \tag{6}$$

- Error rate 3 m (e3m). This metric is similar to the error rates for stereo matching algorithms, but using meters instead of pixels. From all evaluated pixels, we compute the percentage of them where the error is larger than 3 m.
- Error rate 1 m (e1m). This metric works the same way then e3m, but for a stricter margin of 1 m.

5.2. Results SyntCities

We do analyse first the results for the SyntCities. As the data has a synthetic nature, the networks faced a simplified case where a controlled environment was used to render the scenes. Nonetheless, as the ground truth is very accurate, these experiments provided insights about the matching capabilities of the algorithms.

We evaluate the models Stereo_SC, MVS_Stereo_SC and MVS_Full_SC, which were trained on SyntCities and applied the median to fuse all height maps into the final DSM. The results are shown in Table 1. A total of 22 scenes were evaluated and the results are averaged from individual results. Inference for Stereo_SC requires 1.18 s, for MVS_Stereo_SC 0.8 s and for MVS_Full_SC 1.19 s. Times are slightly longer than in the original implementations as the cost volumes are also exported.

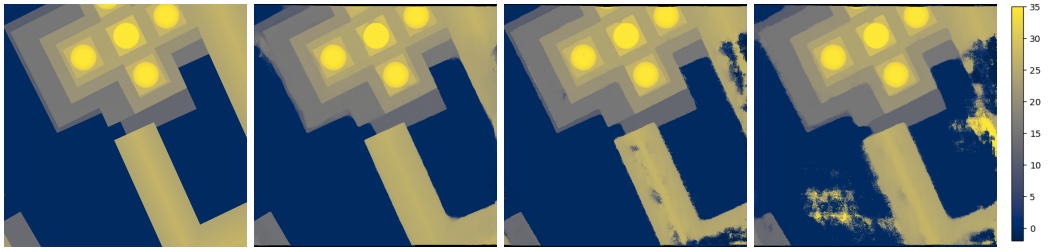
From the presented metrics, we can observe the algorithms achieve a similar performance in the reconstructed DSMs. We show both mean and median based fusions in the results, as the mean one provides information about the presence of outliers in the estimated heights and the median one provides a more robust result. The best performing of the three selected algorithms is Stereo_SC, which is based on AANet. If we analyze e3m, Stereo_SC shows an error rate of 9.38%, which is 1.2% and 2.9% less than MVS_Full_SC and MVS_Stereo_SC respectively, containing less outliers. For the stricter e1m rate, Stereo_SC is again best, with differences of 0.2% and 2.3% in comparison to MVS_Full_SC and MVS_Stereo_SC respectively, showing that MVS_Full_SC has a competitive performance in this metric. With respect to the MAD metric, the results benefit the MVS algorithms. This shows that MVS can achieve a more accurate result for a well matched pixel but the outliers are larger than in the stereo method for areas difficult to match. Regarding MAE and RMSE we also notice a better performance when using the median fusion. For these two metrics the values are consistent with e3m and e1m, observing the best result for Stereo_SC, followed by MVS_Full_SC and MVS_Stereo_SC.

In the Figure 9 there is a visualization for the performance of the evaluated cases. In the upper row, the generated DSMs are compared along with the ground truth, while the lower row shows the absolute error map clipped to a threshold of 1m. The RGB image helps to visualize the texture and geometry of the features to match. As mentioned for the table analysis, the MVS methods present more outliers in areas difficult to match like the texture less areas in the rooftop and ground of the shown building. The Stereo_SC method has less error regions and performs better for the difficult areas. However, around the church domes, the Stereo_SC method is less accurate, especially around boundaries. It is also noticeable how the error regions vary smoothly in the stereo case, whereas for the MVS cases the values vary significantly from one pixel to another. Focusing only on the two MVS results, MVS_Full_SC is better than MVS_Stereo_SC, with a small difference in MAD but a better performance in e3m and e1m.

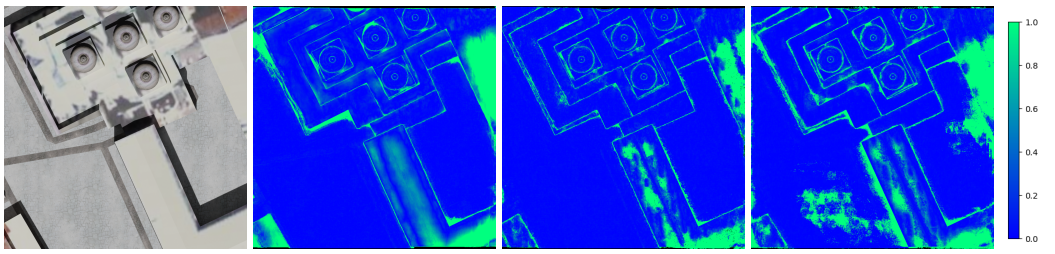
A 3D visualization of the computed DSMs is shown in Figure 10 for the same area as Figure 9. There we can observe how the Stereo_SC method produces smooth areas and the MVS cases suffer from outliers, especially MVS_Stereo_SC, where the values are not even similar to the height range of the scene.

Table 1. DSM generation metrics, based on the fusion of stereo and MVS results for the SyntCities dataset. As indicated by the arrows, the best results are obtained with the lower values of the metrics.

		Metrics				
Network	**Fusion**	**MAD (↓)**	**MAE (↓)**	**RMSE (↓)**	**e3m (↓)**	**e1m (↓)**
Stereo_SC	Mean	0.45	1.26	1.75	11.38	26.22
	Median	0.39	1.14	1.49	9.38	22.12
MVS_Full_SC	Mean	0.32	1.62	1.88	13.04	26.02
	Median	0.29	1.41	1.58	10.55	22.30
MVS_Stereo_SC	Mean	0.39	2.51	2.45	21.23	37.99
	Median	0.29	1.61	1.76	12.27	24.47

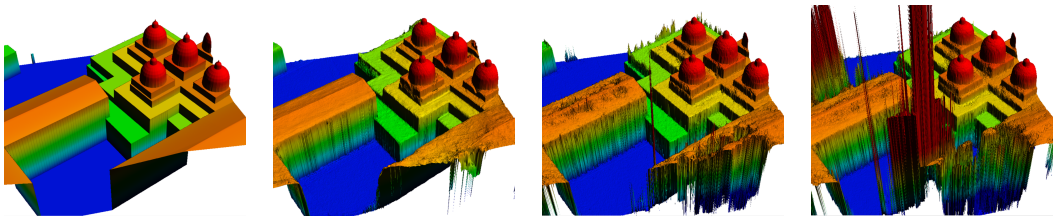


(**a**) DSM-Ground Truth (**b**) Stereo_SC (**c**) MVS_Full_SC (**d**) MVS_Stereo_SC



(**e**) Reference image (**f**) e1m-Stereo_SC (**g**) e1m-MVS_Full_SC (**h**) e1m-MVS_Stereo_SC

Figure 9. DSMs and error maps for a SyntCities sample. For the reference image (**e**) with ground truth (**a**), we show the DSMs computed by using the models Stereo_SC (**b**), MVS_Full_SC (**c**) and MVS_Stereo_SC (**d**). The respective 1 m-error maps (e1m) for the same models are shown in (**f**–**h**). Scale bars for the DSMs and error maps are given as a reference and use meters as unit. Errors are clipped to a maximum of 1 m. Regions in black correspond to undefined pixels by the algorithms.



(**a**) GT (**b**) Stereo_SC (**c**) MVS_Full_SC (**d**) MVS_Stereo_SC

Figure 10. SyntCities computed DSMs, 3D view. For the same perspective given for the ground truth (**a**), we show the results for the models Stereo_SC (**b**), MVS_Full_SC (**c**) and MVS_Stereo_SC (**d**). It covers the same area as the Figure 9. Height values are displayed in blue to red color from low to high.

5.3. Results Dublin

For the experiments applied to the Dublin dataset, we show the obtained results in Table 2. We compare now the models Stereo_Du, MVS_Full_Du and MVS_Stereo_Du, which were finetuned with the Dublin dataset. As this dataset reflect the complexity of real-

world scenes, the performance is lower than the one observed for SyntCities. Inference for Stereo_SC requires 1.27 s, for MVS_Stereo_SC 2.1 s and for MVS_Full_SC 3.03 s. Times are slightly longer than in the original implementations as the cost volumes are also exported and also longer than for SyntCities as many tiles are larger.

Again we observe the results to be in a similar range, demonstrating that all alternatives have reasonable capabilities for the 3D reconstruction. Nonetheless, there are differences to show which one performs best in real data. We observe here that in this case MVS_Full_Du is the leading algorithm followed by Stereo_Du and finally MVS_Stereo_Du. The change about Stereo not leading these results might come from the dataset configuration, as SyntCities was designed to work in a stereo matching framework, rendered already with epipolar geometry.

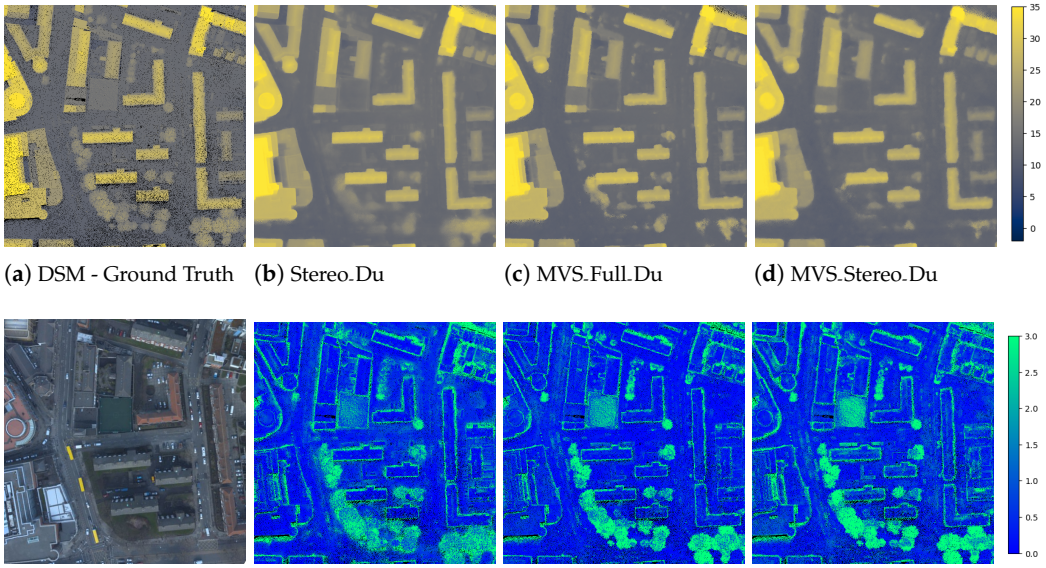
For the e3m rate, MVS_Full_Du leads the table with an advantage of 1.92% and 2.26% over Stereo_Du and MVS_Stereo_Du respectively. A similar trend is observed for the stricter e1m rate, with improvements of 3.51% and 9.12%. The difference in the latter metric is high between both MVS solutions, showing MVS_Full_Du is better than MVS_Stereo_Du by a good margin. Although MVS_Full_Du is also better than Stereo_Du, the difference with respect to stereo is not large, especially for MAD. Focusing on MAD for the median based fusion of each algorithm, Stereo_Du and MVS_Full_Du have only a change of 0.01%, and 0.2% to MVS_Stereo_Du. With respect to the MAE and RMSE, these show a similar trend as MAD. Particularly, RMSE values mean that some outliers present in the Stereo_Du result are larger than for MVS cases.

In Figure 11 we show the results for the computed DSMs. The upper row includes the DSMs and the lower one the error maps, in this case with a threshold of 3 m as the reconstruction is less accurate than for the synthetic data. Still, we observe some similarities to the performance described for SyntCities. The quality around the edges is again better using the MVS algorithms as we can see for buildings and trees. Interestingly, for the trees themselves Stereo_Du achieves a better estimation, as for MVS these areas show errors larger than 3 m. Whilst the metrics are calculated for the entire acquisition track, we only show part of it in the image so that the buildings and edges are zoomed in enough to be easily observed.

Table 2. DSM generation metrics, based on the fused results of stereo and MVS for the Dublin dataset. As indicated by the arrows, the best results are obtained with the lower values of the metrics.

		Metrics				
Network	**Fusion**	**MAD (↓)**	**MAE (↓)**	**RMSE (↓)**	**e3m (↓)**	**e1m (↓)**
Stereo_Du	Mean	2.49	6.06	13.49	47.06	72.68
	Median	0.56	1.92	10.01	15.18	36.76
MVS_Full_Du	Mean	0.60	1.51	2.86	13.97	35.51
	Median	0.55	1.49	2.94	13.26	33.25
MVS_Stereo_Du	Mean	1.1	2.06	3.32	21.20	54.27
	Median	0.75	1.77	3.32	15.52	42.31

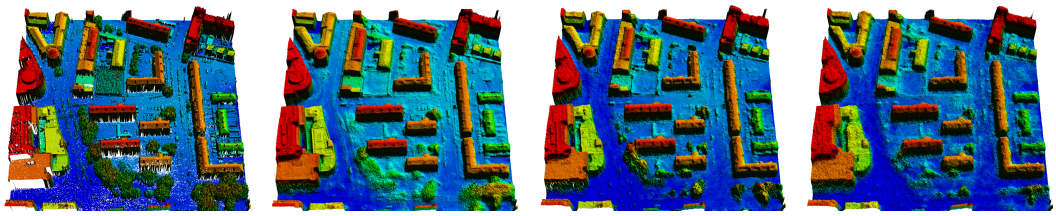
A 3D visualization of the DSMs is displayed in Figure 12. Rooftops are smoother and include less outliers in the Stereo_Du result. Besides, the vegetation is better represented as most of their surface is above ground level comparing with both MVS results. On the other hand, MVS_Stereo_Du and especially MVS_Full_Du compute a better estimation for pixels on the ground level, but they reduce significantly the expected surface for vegetation.



(**a**) DSM - Ground Truth (**b**) Stereo_Du (**c**) MVS_Full_Du (**d**) MVS_Stereo_Du

(**e**) Oblique close image (**f**) e3m-Stereo_Du (**g**) e3m-MVS_Full_Du (**h**) e3m-MVS_Stereo_Du

Figure 11. DSMs and error maps for a Dublin sample. For ground truth (**a**), we show the DSMs computed by using the models Stereo_Du (**b**), MVS_Full_Du (**c**) and MVS_Stereo_Du (**d**). The respective 1 m-error maps (e1m) for the same models are shown in (**f**–**h**). Scale bars in meters for the DSMs and error maps are given as a reference. Errors are clipped to a maximum of 3 m. Regions in black correspond to undefined pixels by the algorithms. The corresponding orthorectified RGB is not shown, as this was not provided in the original dataset for this region. Instead, we show an oblique image captured close to this region in (**e**). This image is not aligned with the results.



(**a**) GT (**b**) Stereo_Du (**c**) MVS_Full_Du (**d**) MVS_Stereo_Du

Figure 12. Dublin computed DSMs, 3D view. For the same perspective given for the ground truth (**a**), we show the results for the models Stereo_Du (**b**), MVS_Full_Du (**c**) and MVS_Stereo_Du (**d**). It covers the same area as the Figure 11.

5.4. Results Confidence

In a separate section, we want to discuss the results of using the confidence values for the fusion with the method presented in Section 4.2. We evaluated the three DSM generation algorithms, namely Stereo_Du, MVS_Full_Du and MVS_Stereo_Du with the same approach, although LAFNet was designed only for stereo data and disparity maps. We studied only the case for the Dublin dataset, as it is more challenging and it has more candidate values for each pixel.

For each of the algorithms we analysed the following cases:

- Optimal: We select the best candidate for each pixel based on the difference with respect to the ground truth. Methods cannot achieve such accuracy, but we use it as a reference of the ideal best performance.
- Mean: We compute the mean of all candidate values to set the height of the pixels as previously used.
- Mean$N$: We remove the N% less confident values for each pixel and then we compute the mean. $N \in \{25, 50\}$

- Median: We compute the median of all candidate values to set the height of the pixels as previously used.
- Median$N$: We remove the N% less confident values for each pixel and then compute the median. $N \in \{25, 50\}$

Since the mean and the median based fusions without removal are the same algorithm as in the previous sections, these values are also found in Table 2. Despite being the median one more robust than the mean case, we include both to give insights about the distribution of the candidate values. The new results are given in Table 3.

With regard to the Stereo_Du case and the mean based fusion, we observe that using the confidence values reduces significantly the presence of outliers. We see that for Mean25 and Mean50 the e3m rate drops to 18.33 and 15.33 respectively from the original 47.06. For the stricter e1m rate, the values drop to 43.03 and 38.69 instead of 72.68. This shows that large outliers were assigned a low confidence value. Considering the median based fusion values, the error rates decrease as well by approximately 2% in both e3m and e1m. By removing significant outliers from the distribution, the MAD of the remaining values gets closer to the ground truth. This is also consistent for MAE and RMSE, where we notice an improvement where the confidence values were used. As the fusion is evaluated per pixel, the algorithm can also be implemented efficiently for parallelization. Hence, the confidence based fusion helps to refine the computed DSM for the stereo case.

Nevertheless, the confidence values do not seem to help in a similar manner the results from MVS_Full_Du and MVS_Stereo_Du. If we focus on the MVS_Full_Du case, we observe that the higher the percentage of removed pixels, the higher the error rate as well. Although the difference is small, we note that there is no trend towards improvement. Addressing the MVS_Stereo_Du case, we notice for both mean and median based fusions a slightly better performance by using $rem_{\%} = 25$ in all metrics. By setting $rem_{\%} = 50$ the error rate is not decreasing. As LAFNet was developed for a distinct input data, we consider many aspects should be taken into account to redesign the network to handle depth maps as well. Some of these aspects include:

- Disparity maps and images are both in pixels and work in a 2D domain, while depth is meters and represents a 3D space, which is harder to correlate with the input images without the homography matrix information. Besides, depth and disparity ranges are inversely proportional and span different numerical ranges.
- Cost volumes used in UniMVSNet have a downsampling rate of $\times 4$, which means the number of pixels is 1/16 of the original image size, missing details while upsampling the cost volume to be used by LAFNet. Nonetheless, the memory demands of the MVS algorithms limit the size of the cost volume to be computed.
- The learned features for the cost volumes vary from those for stereo matching. Especially for the MVS_Full_Du case, where many views are taken into account, the features for a reference image contain information from many additional views, where not all pixels are always visible. MVS_Stereo_Du seems to suffer less from this effect.
- MVS algorithms already make a fusion from different views based on the learned weights. Hence, the confidence might not be so discriminative to filter bad candidates in the estimated map.

The design of a new confidence network is out of our scope, but after studying the effect on the stereo data, we see potential to use the confidence based fusion as a good strategy to create DSMs.

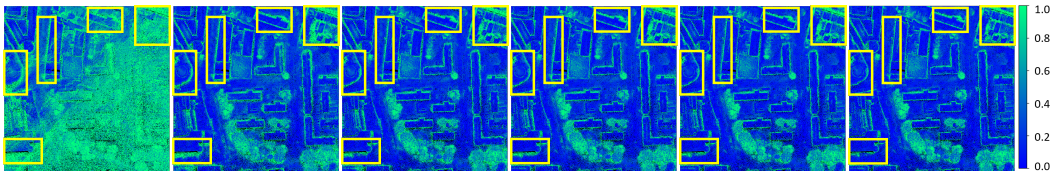
We show visually the results of the stereo case by using different $rem_{\%}$ rates. In Figure 13 the images show the impact of the confidence based fusion. For the mean fusion cases, we see a significant reduction of the error rate, particularly between no confidence

guidance and Mean25, it also improves the fusion around edges for the Mean50 result. The median fusion is more robust and as shown in (d) is less influenced by outliers. By using the confidence values, the fusion improves again mostly around building edges. As observed for the results of the stereo method, these areas are challenging for AANet, but with this guided fusion we can improve the accuracy of the computed DSM.

A 3D representation for the same area is shown in Figure 14. Improvements are mostly in the edges of buildings (smoother in the median cases with confidence), less artifacts on the ground level (excluding cars). Regions highlighted in Figure 13 can also be compared for the 3D representation to observe changes.

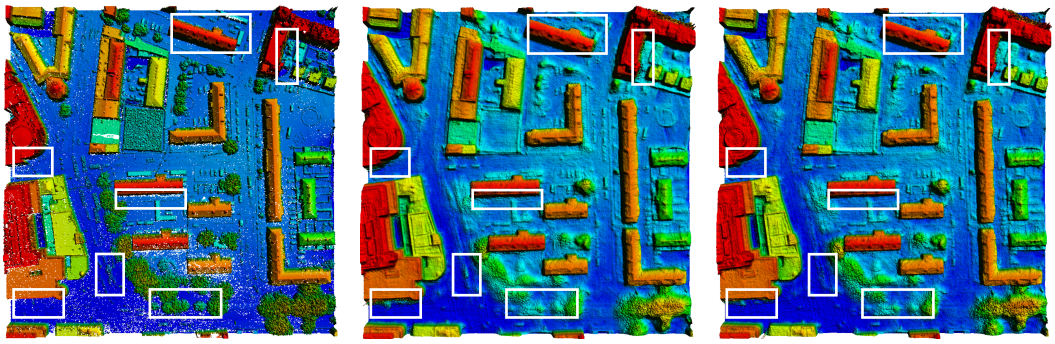
Table 3. DSM generation metrics, based on the fusion of stereo and MVS results for the Dublin dataset. In this case, the confidence was used for the fusion process. As indicated by the arrows, the best results are obtained with the lower values of the metrics.

		Metrics				
Network	**Fusion**	**MAD (↓)**	**MAE (↓)**	**RMSE (↓)**	**e3m (↓)**	**e1m (↓)**
Stereo_Du	Optimal	0.06	0.57	5.48	4.00	10.47
	Mean	2.49	6.06	13.49	47.06	72.68
	Mean25	0.67	2.30	8.95	18.33	43.03
	Mean50	0.59	1.83	7.04	15.33	38.69
	Median	0.56	1.92	10.01	15.18	36.76
	Median25	0.53	1.84	9.82	14.57	34.88
	Median50	0.53	1.69	7.99	13.79	34.12
MVS_Full_Du	Optimal	0.14	0.71	1.96	6.04	14.82
	Mean	0.60	1.51	2.86	13.97	35.51
	Mean25	0.60	1.51	2.88	13.96	35.23
	Mean50	0.63	1.55	2.95	14.25	36.27
	Median	0.55	1.49	2.94	13.26	33.25
	Median25	0.57	1.51	2.96	13.40	34.00
	Median50	0.62	1.57	3.03	13.84	36.50
MVS_Stereo_Du	Optimal	0.09	0.33	1.04	1.89	6.58
	Mean	1.10	2.06	3.32	21.20	54.27
	Mean25	1.04	1.99	3.28	19.72	51.97
	Mean50	1.08	2.06	3.42	20.21	52.90
	Median	0.75	1.77	3.32	15.52	42.31
	Median25	0.76	1.78	3.36	15.63	42.15
	Median50	0.84	1.89	3.52	16.49	45.32



(**a**) Mean (**b**) Mean25 (**c**) Mean50 (**d**) Median (**e**) Median25 (**f**) Median50

Figure 13. Dublin DSMs created with confidence based fusion - Stereo case. We show cases for mean fusion without confidence (**a**), with $rem_{\%} = 25$ (**b**) and with $rem_{\%} = 50$ (**c**). Similar cases are presented for the median in (**d**–**f**). Scale bar for the error is given in meters. Yellow rectangles highlight areas with significant differences.



(**a**) Ground Truth (**b**) Median (**c**) Median50

Figure 14. Generated DSMs for a Dublin region in a 3D representation—Stereo case. Region is the same as for Figure 13. We show three DSMs: ground truth, median fusion (no confidence based) and median fusion $rem_{\%} = 50$. Changes are highlighted with the white rectangles.

6. Conclusions

We presented in this paper a comparison between stereo and multi-view stereo (MVS) deep learning algorithms. From the presented results, we show how all solutions (Stereo, MVS_Full and MVS_Stereo) were able to compute a reliable DSM and preserving most of the geometric information. Stereo produces smoother results and is less prone to outliers, facing challenges in areas adjacent to edges. On the other hand, MVS_Full and MVS_Stereo provide a better height estimation for those areas where the matching is not so challenging, but it also suffer from larger outliers where the matching fails, including textureless areas. We consider MVS_Full to be the most robust solution, also due to the low MAD values. Stereo also shows a good performance and benefits more from context information to compute a similar estimation for regions belonging to the same object, presenting errors mostly on edges instead. MVS_Stereo showed the lowest performance between the three approaches, leading to larger outliers and less accuracy for the strict e1m rate. Between the two basic fusion algorithms, we find that median fusion is superior to mean fusion in all cases, so we do not recommend the latter as it is not robust to the influence of large outliers present in the estimated heights.

Regarding the confidence based fusion strategy we adopted, the results for the Stereo method showed an improvement, particularly for areas adjacent to the edges where the matching algorithm is prone to errors, compensating this flaw. However, the same method did not lead to more accurate DSMs for the MVS_Full and MVS_Stereo algorithms. We described some factors that could explain this issue, such as the discrepancies between depth and disparity maps, and the cost volumes sizes.

We additionally provide a processed version of the Dublin dataset to be applied in stereo and MVS algorithms, encouraging the community to continue the experiments in this direction or to easily apply the new architectures in the remote sensing field.

Future Work

Based on the obtained results, we observed that the confidence based fusion lead to good results in the height maps estimated by the stereo algorithm. We would like to explore possible changes to the network to obtain also a good performance for the MVS cases.

Additionally, a more sophisticated algorithm using the confidence values to fuse the DSM should be explored, not only the removal of bad pixels and the median of the remaining values. A neural network that uses both height and confidence maps as inputs for the fusion could be an interesting research topic.

Author Contributions: Conceptualization, M.F.R., P.d. and F.F.; methodology, M.F.R., P.d. and F.F.; software, M.F.R. and P.d.; validation, M.F.R., P.d. and F.F.; data curation, M.F.R. and P.d.; writing—original draft preparation, M.F.R.; writing—review and editing, P.d. and F.F.; supervision, P.d. and F.F. All authors have read and agreed to the published version of the manuscript.

Funding: Mario Fuentes Reyes was funded by a DAAD-DLR Research Fellowship (No. 57478193) to pursue his PhD studies.

Data Availability Statement: The data used in this article is open to the public and can be downloaded from the respective websites. SyntCities can be downloaded from https://zenodo.org/records/6967325, the original Dublin dataset is available at https://geo.nyu.edu/?f%5Bdct_isPartOf_sm%5D%5B%5D=2015+Dublin+LiDAR (accessed on 20 December 2024) and the new processed Dublin dataset is stored at https://zenodo.org/records/12772927 (accessed on 20 December 2024).

Acknowledgments: We thank the authors of the Dublin dataset for providing such a large and good quality data in free access mode.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional Neural Network
DSM	Digital Surface Model
e1m	Error rate 1 m
e3m	Error rate 3 m
FCN	Fully Connected Networks
GRU	Gated Recurrent Unit
GSD	Ground Sample Distance
MAD	Median Absolute Deviation
MVS	Multi-view Stereo
SAR	Synthetic Aperture Radar
SGM	Semi-Global Matching

References

1. Fuentes Reyes, M.; d'Angelo, P.; Fraundorfer, F. An evaluation of stereo and multiview algorithms for 3D reconstruction with synthetic data. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2023**, *48*, 1021–1028. [CrossRef]
2. Fuentes Reyes, M.; D'Angelo, P.; Fraundorfer, F. SyntCities: A Large Synthetic Remote Sensing Dataset for Disparity Estimation. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 10087–10098. [CrossRef]
3. Laefer, D.F.; Abuwarda, S.; Vo, A.V.; Truong-Hong, L.; Gharibi, H. 2015 Aerial Laser and Photogrammetry Survey of Dublin City Collection Record. 2017. Available online: http://hdl.handle.net/2451/38684 (accessed on 20 December 2024).
4. d'Angelo, P.; Kuschk, G. Dense multi-view stereo from satellite imagery. In Proceedings of the 2012 IEEE International Geoscience and Remote Sensing Symposium, Munich, Germany, 22–27 July 2012; pp. 6944–6947.
5. Scharstein, D.; Szeliski, R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comput. Vis.* **2002**, *47*, 7–42. [CrossRef]
6. Hirschmuller, H. Stereo Processing by Semiglobal Matching and Mutual Information. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 328–341. [CrossRef] [PubMed]
7. Zbontar, J.; LeCun, Y. Stereo matching by training a convolutional neural network to compare image patches. *J. Mach. Learn. Res.* **2016**, *17*, 1–32.
8. Mayer, N.; Ilg, E.; Hausser, P.; Fischer, P.; Cremers, D.; Dosovitskiy, A.; Brox, T. A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4040–4048.
9. Kendall, A.; Martirosyan, H.; Dasgupta, S.; Henry, P.; Kennedy, R.; Bachrach, A.; Bry, A. End-to-end learning of geometry and context for deep stereo regression. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 66–75.

10. Zhang, F.; Prisacariu, V.; Yang, R.; Torr, P.H. GA-Net: Guided Aggregation Net for End-to-End Stereo Matching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 185–194.
11. Xu, H.; Zhang, J. AANet: Adaptive Aggregation Network for Efficient Stereo Matching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 1959–1968.
12. Zhang, F.; Qi, X.; Yang, R.; Prisacariu, V.; Wah, B.; Torr, P. Domain-invariant stereo matching networks. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part II 16; Springer: Cham, Switzerland, 2020; pp. 420–439.
13. Lipson, L.; Teed, Z.; Deng, J. RAFT-Stereo: Multilevel recurrent field transforms for stereo matching. In Proceedings of the 2021 International Conference on 3D Vision, London, UK, 1–3 December 2021; pp. 218–227.
14. Li, Z.; Liu, X.; Drenkow, N.; Ding, A.; Creighton, F.X.; Taylor, R.H.; Unberath, M. Revisiting Stereo Depth Estimation From a Sequence-to-Sequence Perspective with Transformers. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 6177–6186.
15. Xu, H.; Zhang, J.; Cai, J.; Rezatofighi, H.; Yu, F.; Tao, D.; Geiger, A. Unifying Flow, Stereo and Depth Estimation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 13941–13958. [CrossRef]
16. Su, W.; Tao, W. Efficient Edge-Preserving Multi-View Stereo Network for Depth Estimation. *Proc. AAAI Conf. Artif. Intell.* **2023**, *37*, 2348–2356. [CrossRef]
17. Wang, X.; Xu, G.; Jia, H.; Yang, X. Selective-Stereo: Adaptive Frequency Information Selection for Stereo Matching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 16–22 June 2024; pp. 19701–19710.
18. Schönberger, J.L.; Zheng, E.; Pollefeys, M.; Frahm, J.M. Pixelwise View Selection for Unstructured Multi-View Stereo. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–16 October 2016.
19. Galliani, S.; Lasinger, K.; Schindler, K. Massively Parallel Multiview Stereopsis by Surface Normal Diffusion. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 873–881.
20. Yao, Y.; Luo, Z.; Li, S.; Fang, T.; Quan, L. MVSNet: Depth Inference for Unstructured Multi-view Stereo. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
21. Yao, Y.; Luo, Z.; Li, S.; Shen, T.; Fang, T.; Quan, L. Recurrent MVSNet for High-resolution Multi-view Stereo Depth Inference. In Proceedings of the Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
22. Gu, X.; Fan, Z.; Zhu, S.; Dai, Z.; Tan, F.; Tan, P. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 2495–2504.
23. Zhang, J.; Yao, Y.; Li, S.; Luo, Z.; Fang, T. Visibility-aware Multi-view Stereo Network. *Br. Mach. Vis. Conf. (BMVC)* **2020**, *131*, 199–214.
24. Zhang, Y.; Zhu, J.; Lin, L. Multi-View Stereo Representation Revist: Region-Aware MVSNet. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 17376–17385.
25. Xiong, K.; Peng, R.; Zhang, Z.; Feng, T.; Jiao, J.; Gao, F.; Wang, R. CL-MVSNet: Unsupervised Multi-View Stereo with Dual-Level Contrastive Learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023; pp. 3769–3780.
26. Zhang, Z.; Peng, R.; Hu, Y.; Wang, R. GeoMVSNet: Learning Multi-View Stereo With Geometry Perception. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 21508–21518.
27. Vats, V.K.; Joshi, S.; Crandall, D.J.; Reza, M.A.; Jung, S.H. GC-MVSNet: Multi-View, Multi-Scale, Geometrically-Consistent Multi-View Stereo. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 3–8 January 2024; pp. 3242–3252.
28. Peng, R.; Wang, R.; Wang, Z.; Lai, Y.; Wang, R. Rethinking Depth Estimation for Multi-View Stereo: A Unified Representation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022.
29. Hu, X.; Mordohai, P. A Quantitative Evaluation of Confidence Measures for Stereo Vision. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2121–2133. [PubMed]
30. Poggi, M.; Tosi, F.; Mattoccia, S. Quantitative Evaluation of Confidence Measures in a Machine Learning World. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
31. Poggi, M.; Mattoccia, S. Learning from scratch a confidence measure. In Proceedings of the British Machine Vision Conference (BMVC), York, UK, 19–22 September 2016; pp. 1–12.
32. Seki, A.; Pollefeys, M. Patch Based Confidence Prediction for Dense Disparity Map. In Proceedings of the British Machine Vision Conference (BMVC), York, UK, 19–22 September 2016; p. 23.

33. Poggi, M.; Mattoccia, S. Learning to Predict Stereo Reliability Enforcing Local Consistency of Confidence Maps. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4541–4550.
34. Kim, S.; Min, D.; Kim, S.; Sohn, K. Unified Confidence Estimation Networks for Robust Stereo Matching. *IEEE Trans. Image Process.* **2019**, *28*, 1299–1313. [CrossRef] [PubMed]
35. Kim, S.; Kim, S.; Min, D.; Sohn, K. LAF-Net: Locally Adaptive Fusion Networks For Stereo Confidence Estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
36. Denninger, M.; Sundermeyer, M.; Winkelbauer, D.; Olefir, D.; Hodan, T.; Zidan, Y.; Elbadrawy, M.; Knauer, M.; Katam, H.; Lodhi, A. Blenderproc: Reducing the reality gap with photorealistic rendering. In Proceedings of the International Conference on Robotics: Sciene and Systems, RSS 2020, Virtual, 12–16 July 2020.
37. Fusiello, A.; Trucco, E.; Verri, A. A compact algorithm for rectification of stereo pairs. *Mach. Vis. Appl.* **2000**, *12*, 16–22. [CrossRef]
38. Höhle, J.; Höhle, M. Accuracy assessment of digital elevation models by means of robust statistical methods. *ISPRS J. Photogramm. Remote Sens.* **2009**, *64*, 398–406. [CrossRef]