DSM2DTM: AN END-TO-END DEEP LEARNING APPROACH FOR DIGITAL TERRAIN MODEL GENERATION

K. Bittner[a,*], S. Zorzi[b], T. Krauß[a], P. d'Angelo[a]

[a]Remote Sensing Technology Institute, German Aerospace Center (DLR), Weßling, Germany –
(ksenia.bittner, thomas.krauss, pablo.angelo)@dlr.de
[b]Institute of Computer Graphics and Vision, Graz University of Technology, Austria – stefano.zorzi@icg.tugraz.at

KEY WORDS: Digital Surface Model, Digital Terrain Model, Deep Learning, Ground Filtering, Fully Convolutional Networks.

ABSTRACT:

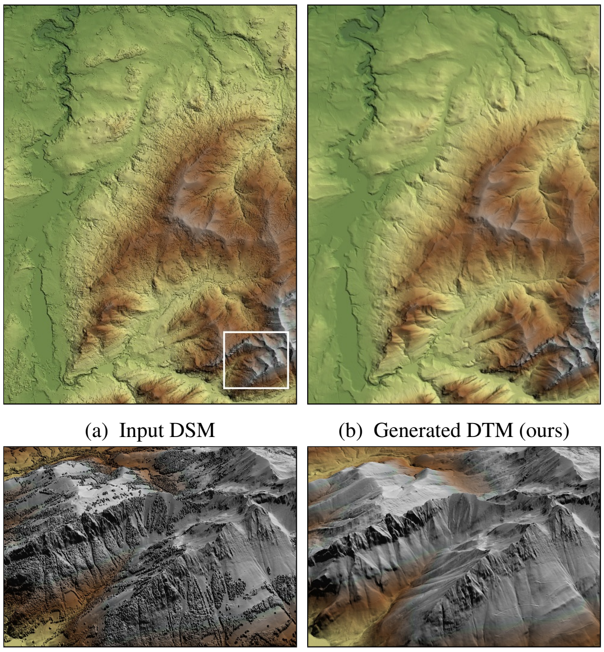
Remotely sensed Earth elevation data or digital surface model (DSM) typically contains both terrain and above-ground information such as vegetation and man-made constructions. However, many applications require pure bare-terrain data, also known as digital terrain model (DTM). But how do we separate 3D objects on the DSM from the ground? The most commonly used approaches are still based on various filtering techniques, which in turn involve the pre-definition of thresholds or specific parameters depending on the inhomogeneity of the scene. Despite many long existing and newly developed approaches the general fully automatic extraction of large-scale, reliable DTMs is still a problem – especially the preservation of steep terrain features in terraced landscapes. In this context, we explore several deep learning models and select one based on the EfficientNet architecture. This model serves as an encoder in the UNet-shaped framework and – despite its relatively low amount of parameters compared to common network architectures – it can automatically distinguish non-ground pixels and estimate the bare-ground height information while maintaining the complexity of the anthropogenic geomorphology of landscapes. In a series of experiments, we demonstrate that the DTM generated with the proposed method significantly outperforms other DTM generation approaches – both quantitatively and qualitatively. To enable further comparisons with our methodology the training, validation and test datasets have been collected together and made available at https://github.com/KseniaBittner/DSM2DTM.

1. INTRODUCTION

A Digital terrain model (DTM) is a representation of the bare Earth surface without vegetation and any human constructions such as buildings, roads, bridges, and others. A DTM is a powerful supportive information in various disciplines such as surveying and construction engineering of pipelines, canals or highways, disaster management systems, water-runoff, land cover mapping and many more. Therefore, having a precisely accurate, detailed, and not over-smoothed DTM is a requirement for developing technologies.

DTMs can be generated directly from terrain measurements or extracted from digital surface models (DSMs). DSMs can, in turn, be derived from active sensing approaches like laser scanning, radar interferometry, or from processing optical stereo images from either aerial or satellite sensors (Krauß et al., 2011). Deriving a DTM from a given DSM necessitates the detection of all above-ground objects first, followed by their removal, and then interpolating the resulting empty spaces with meaningful height information. In addition to a range of classical DSM filtering algorithms, some deep learning based methodologies have also been recently developed. However, most of those methods, are multi-step procedures which often require predefined conditions, filter characteristics, or thresholds. Moreover, a common issue with existing algorithms is their failure to preserve sharp terrain slopes, especially in the terraced landscapes.

In this paper, we propose a deep learning approach capable to automatically generate a large-scale DTM out of a provided


(a) Input DSM (b) Generated DTM (ours)

(c) Zoomed input DSM (d) Zoomed DTM (ours)

Figure 1. Sample of area from our Fribourg test dataset illustrated both (a) input DSM to the network and (b) the resulting DTM. Zoomed areas, highlighted in a white box, in a zone with steep mountain reliefs on both DSM and DTM are depicted in (c) and (d), respectively.

[*] Corresponding author

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-1-W1-2023-925-2023 | © Author(s) 2023. CC BY 4.0 License.

DSM in an end-to-end manner. The approach is independent of the type of terrain and works equally good on mountain and urban scenes (see Figure 1).

2. RELATED WORK

A common procedure for generating a DTM from remote sensing data involves removing above-ground objects (e.g. buildings, vegetation, and vehicles) and replacing the eliminated pixels with appropriate elevation values. However, the main challenge lies in detecting these above-ground objects quickly and accurately. To address this problem, a wide variety of rule-based algorithms have been developed.

Some rules work through morphological operations, making an assumption that terrain does not contain sharp height differences within a local neighborhood (Vosselman, 2000; Sithole and Vosselman, 2001; Wang and Tseng, 2010). Looking within a certain size of window, a predefined threshold is used for filtering points with distinctively different heights. Some approaches rely on automatic threshold definitions, such as adaptive filters (Sithole and Vosselman, 2001) or knowledge-based filters (Wang and Shan, 2009). But these strategies are still suitable only for relatively flat terrains. Moreover, the selection of an appropriate size of search neighbourhood plays a crucial role. When the search neighbourhood is too small, only small features can be detected well but large objects are then marked as ground. If the search neighbourhood is too big, peaks, rocks, and hills can be filtered out (Pingel et al., 2013). To overcome those problems, Arefi et al. (2007) developed an iterative filtering method based on geodetic dilation operator which thresholds off-terrain information in an adaptive window size. Krauß and Reinartz (2010) propose a steep edge detection approach which applies two median filters with different filter sizes to generate a DTM. So areas at the bottom of steep walls are detected. But there are some drawbacks like the identification of objects on the rooftops as independent buildings resulting in incorrect detection of lower roof pixels as ground pixels or overseeing small bushes and setting them as ground pixels. For better preservation of steep slopes and local heterogeneity of elevations, Hui et al. (2016) combines a progressive morphological filtering algorithm with multi-level interpolation filtering approach. In an iterative manner with a gradually downsized filtering window, a morphological opening operation is applied to detect off-terrain pixels, while kriging interpolation is used to interpolate eliminated pixels at different levels according to the different search neighbourhood.

More advanced methodologies pay more attention on overcoming window size sensitivity problem and parameterization minimization for automatic DTM-from-DSM generation. The method of Duan et al. (2019) uses rule-based classification utilizing multi-scale morphological analysis to detect above-ground objects while preserving local reliefs in both flat and highly mountainous areas. Defining several different thresholds based on statistics of bare-terrain elevations from input DSM, a final DTM is generated via the least squares solution. Authors assure that the method is robust to worldwide large-scale DSMs and does not require parameters tuning. Since the slope aspect is often not entirely mono-directional across hillslope, the work of Pijl et al. (2020) proposed non-linear filters driven from terrain slope anisotropy which has a primary focus on the preservation of sharp terrain features.

In recent years, the rapid development of deep learning techniques and especially their impressive performance on classification and segmentation tasks, shifted the focus towards learning-based methods for DTM generation problems as well. For example, Marmanis et al. (2015) explored the performance of multi-layer perceptron (MLP) and its knowledge-transferability on satellite images from different sensors to separate ground/off-ground information. The method demonstrate good performance in dense urban areas and its independence from any predefined thresholds. Following this study, Tapper (2016) investigates a three-layer artificial neural network (ANN), which simultaneously extract features from RGB, DSM and NIR images to output four classes: ground, man-made objects, vegetation, and water. Then the DTM height is calculated for all detected and filtered above-ground objects. Later, Gevaert et al. (2018) used more advanced architecture based on convolutional neural network (CNN) to generate a DTM by firstly extracting off-ground training samples through morphological operations and then training the developed model. Although, the learning-based approaches for classification task methodologies demonstrate superior performances in comparison to traditional filter-based approaches, they are still two-step strategies.

The power of deep neural networks is not limited to detection, classification or segmentation tasks. With modern architectures it is also possible to generate depth images – images with height values. A recent study of Amirkolaee et al. (2022) demonstrates the potential of deep learning to automatically generate DTMs out of DSMs using a UNet encoder-decoder architecture with residual connections. However, some necessary pre- and post-processing steps are still involved in the procedure. The authors perform a pre-processing of the input DSM by localizing the prepared for training height images in pre-defined height ranges. As post-processing step, a multi-scale fusion strategy is involved to produce the final DTM from generated DTMs at different scales and with different spatial shifts.

Despite significant contributions from deep learning methodologies towards DTM generation, none of the approaches are end-to-end, requiring several additional steps to generate the final DTM. Different from them, in this paper we present a method which is able to overcome the problem of multi-stage procedure and generate a DTM from DSM in a single step. Our contributions are:

- We developed a method based on a simple and efficient network architecture, which has far less amount of parameters in comparison to most popularly used network architectures.
- We perform a comparison study between most common deep network architectures and demonstrate that the model do not need to be very complex and have a huge amount of parameters in order to fulfill a task of generating DTMs out of DSMs.
- We demonstrate that no pre-processing steps of input data are needed. With a smart data normalization procedure during the training, the performance of the network is not limited due to the different regions with various height ranges.
- Our method is independent of any predefined thresholds and can generalize even over particularly challenging terrain types like steep slopes, vegetated slopes, or discontinuous terrain features.

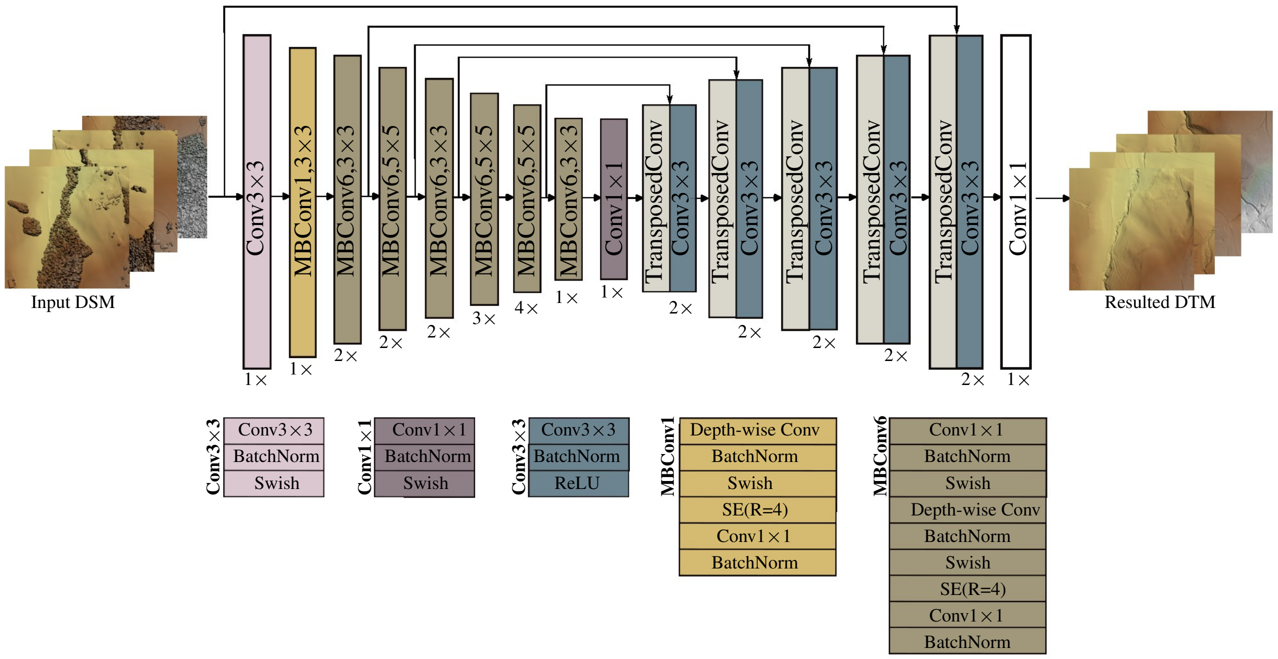


Figure 2. Schematic overview of the proposed architecture for DTM generation out of DSM. The module is represented by an encoder-decoder network which is based on EfficientNet encoder for analysing the surface information on detecting above-ground areas, and a decoder module, which generate a clean terrain landscape.

3. METHODOLOGY

Recent studies of Bittner et al. (2020) and Stucker and Schindler (2020) demonstrated that CNNs are capable of generating good-quality height images representing urban constructions from photogrammetric DSMs. Inspired by those studies and that the interest to deep-learning approaches for DSM-to-DTM rapidly increases, we developed an end-to-end methodology which is capable to generate DTMs out of DSMs even for complex mountain and urban scenes. Among widely used backbone networks, such as UNet (Ronneberger et al., 2015) or ResNet (He et al., 2016), we have decided to use EfficientNet (Tan and Le, 2019) and implement it in a UNet shape architecture.

3.1 Network Architecture

In order to achieve better accuracy, many developed baseline networks, are usually further scaled up by one of three dimensions – depth (He et al., 2016), width (Szegedy et al., 2015), or image size (Huang et al., 2019). In general, the scaling is performed arbitrarily and, as a result, requires tedious manual tuning that is in turn very time-consuming. The research of Tan and Le (2019) proposes to scale the network not only in one dimension but in all three dimensions uniformly balancing them with a constant ratio. Their baseline network called EfficientNet is constructed using a multi-objective neural architecture search, which finds an optimal input size, depth, and width for the architecture. The basic building block used to design a network is borrowed from the class of efficient mobile-size models and has a mobile inverted residual structure called MBConv (Sandler et al., 2018; Tan et al., 2019). In contrary to a traditional residual block, the input and output of the MBConv block are thin bottleneck layers which are joint via a short skip connection. In between the input is extended using a 1×1 convolution, then narrowed by a 3×3 lightweight depth-wise convolution to filter features as a source of non-linearity, and finally squeezed by another 1×1 convolution in order to match the initial number of channels. As a result the inverted residual block has far fewer parameters in comparison to the original residual block which fits exactly the purpose of mobile networks. Moreover, Tan and Le (2019) attach asqueeze-and-excitation (SE) block (Hu et al., 2018) with reduction ratio (R) of four to MBConv block to add a content-aware mechanism to explicitly model channel relationships and channel interdependencies. Besides, they use a Swish activation function $f_{Swish} = \frac{x}{1+e^{-\beta x}}$, which combines the power of both ReLU and LeakyReLU activation functions, but is learnable due to the $\beta \geq 0$ parameter.

There exist a family of EfficientNets which are based on several popular ConvNet architectures. In our work we use a baseline model EfficientNet-B0 as an encoder for a UNet shape architecture. We have a five level decoder, where each decoder construction block consists of a transposed convolution, which performs a learnable up-sampling of the features, twice followed by a sequence of 3×3 conv – batch norm – ReLU. The number of parameters for overall architecture is 14.115 million. Figure 2 depicts the detailed network architecture.

3.2 Loss Function

To generate an image with detailed terrain information, we apply an absolute error loss $\mathcal{L}_1$ during training

$$\mathcal{L}_1(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_1, \tag{1}$$

to compute the absolute difference between an estimated set of DTM samples $\hat{\boldsymbol{y}}$ and the actual, ground truth DTM set of samples $\boldsymbol{y}$.

3.3 Baselines

We compare the developed DSM-to-DTM approach against a non deep learning approach based on the work of Krauß et al. (2008) and two baseline networks.

Filtering approach: Instead of applying a classical morphological erosion using a filter size of estimated largest cross-section of all buildings for their elimination from DSMs (Weidner and Förstner, 1995), the approach of Krauß et al. (2008) propose to use percentile filters instead of the gray-value mophological filter which is less sensitive to outliers below the real terrain in a DSMs. Using a low percentile filter resembling a morphologic erosion followed by a high percentile filtering resembling a morphologic dilation results in a percentile opening of the DSM. Afterwards a gaussian filtering using the same filter size is applied to obtain a smoother DTM. The method is applicable for the whole image at ones without tiling.

UNet: We found it reasonable to compare the results with a fairly standard UNet (Ronneberger et al., 2015) architecture, since the general shape of our developed network also follows its form. The last layer of the network is a 1×1 convolution which outputs one channel image – the DTM. The number of parameters for the overall architecture is 34.526 million.

UResNet: In remote sensing it is very common to place popular ConvNet architectures developed for classification tasks into a UNet form (Bittner et al., 2020; Schuegraf et al., 2022). A common nowadays ResNet50 build in UNet shape is taken as a baseline to compare our results with. In this network, the last layer is also set as a 1×1 convolution which outputs one channel image – a DTM. The number of parameters for the overall architecture is 139.421 million.

4. EXPERIMENTS

4.1 Dataset

We evaluated our method on the dataset consisting of rasterized DSMs and DTMs with a ground sampling distance (GSD) of 0.5 m. The data is provided by the Federal Office of Topography of Switzerland and is freely available on the Swisstopo Portal[1]. According to an official description the DSMs are derived from airborne light detection and ranging (LiDAR) using all relevant returns filtered by the spike-free algorithm (Khosravipour et al., 2016). For deriving the DTMs the airborne LiDAR data were used for areas below 2000 m and automatic stereo-matching techniques were used for areas above 2000 m. In addition, the remaining gaps were closed with manual stereo-matching in case the automatic stereo-matching did not work.

We used the data of the Cantons of Zuerich, St. Gallen, and Vaud for training and validation. The data for each Canton is provided in 2000×2000 px image patches out of which we have selected only eight random non-overlap samples of size 256×256 px. We took 10 % of images from each of those Cantons to perform a validation phase. The Canton of Fribourg of 588 km^2 area was used for testing. The exact data distribution between the training and validation samples and the exact images tiling on patches is accessible at https://github.com/KseniaBittner/DSM2DTM.

4.2 Implementation Details

We have implemented the DSM-to-DTM pipeline in *PyTorch* and run it on a single NVIDIA TITAN X (PASCAL) GPU with 12 GB of memory. We use training and validation patches of size 256×256 px for three selected Cantons. The prepared training dataset for the learning process consisted of 44 580 pairs of samples and validation dataset contained 1241 pairs of samples. During the training phase the samples were augmented not only by random horizontal and vertical flipping but also by random rotations to improve the robustness of the model. To facilitate the network optimization process and maximize the probability of obtaining good results, we performed the normalization of DSM and DTM data for neural network training. We followed the strategy of Stucker and Schindler (2020) and globally normalised terrain heights by centering them to mean height 0 and scaling by the global standard deviation of the heights, computed on all patches of DSM data from the training set and averaged afterwards.

The network is trained in a fully supervised manner by minimizing the pixel-wise absolute distance between generated DTM and the ground truth. We employed the mini-batch stochastic gradient descent (SGD) using the Adam optimizer (Kingma and Ba, 2014) with an initial learning rate of $\alpha = 0.0002$ which was dropped by a factor of 10 after 100 epochs. The momentum parameters were set to $\beta_1 = 0.5$ and $\beta_2 = 0.999$, and a batch size of 16 was used. The training was performed for a total of 200 epochs.

At inference time, we do a large-scale DTM generation by applying only the best performing models from the validation stage in a sliding window of size 256×256 px with overlap of 128 px between neighboring patches. The resulting DTM is averaged at the overlapping regions.

4.3 Evaluation Metrics

For quantitative evaluation of resulted DTM, we measure the mean absolute error (MAE)

$$\varepsilon_{\text{MAE}}(\boldsymbol{h}, \hat{\boldsymbol{h}}) = \frac{1}{n} \sum_{j=1}^{n} (|\hat{h}_j - h_j|) \tag{2}$$

the root mean squared error (RMSE)

$$\varepsilon_{\text{RMSE}}(\boldsymbol{h}, \hat{\boldsymbol{h}}) = \sqrt{\frac{1}{n} \sum_{j=1}^{n} (\hat{h}_j - h_j)^2}, \tag{3}$$

the median residual error (MedErr)

$$\varepsilon_{\text{MedErr}}(\boldsymbol{h}, \hat{\boldsymbol{h}}) = \text{median}(\hat{h}_j - h_j), \tag{4}$$

and the normalized median absolute deviation (NMAD)

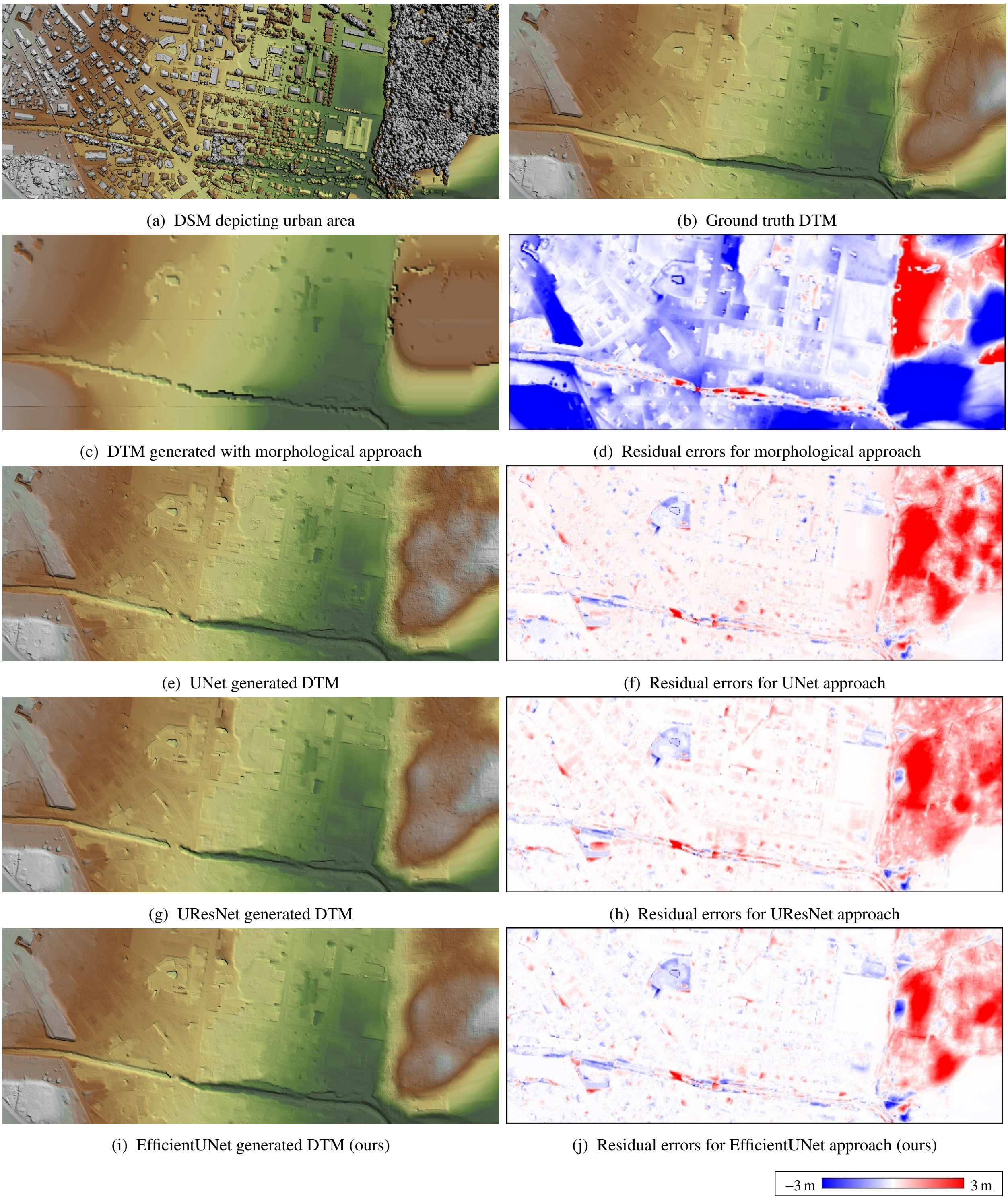
$$\varepsilon_{\text{NMAD}}(\boldsymbol{h}, \hat{\boldsymbol{h}}) = 1.4826 \cdot \underset{j}{\text{median}}(|\Delta h_j - m_{\Delta \boldsymbol{h}}|), \tag{5}$$

computed pixel-wise between predicted $\hat{\boldsymbol{h}}$ and reference $\boldsymbol{h}$ heights for a total number of n pixels. Moreover, for NMAD we denote a height errors as $\Delta h_j = \hat{h}_j - h_j$ and median error as $m_{\Delta \boldsymbol{h}} = median(\hat{h}_j - h_j)$.

5. RESULTS AND DISCUSSION

We have performed the evaluation on Fribourg city and its vicinity. This area combines both the urban environment and very steep mountain neighborhood. This complex area is selected for

[1] https://www.swisstopo.admin.ch/en/geodata.html


(a) DSM depicting urban area

(b) Ground truth DTM

(c) DTM generated with morphological approach

(d) Residual errors for morphological approach

(e) UNet generated DTM

(f) Residual errors for UNet approach

(g) UResNet generated DTM

(h) Residual errors for UResNet approach

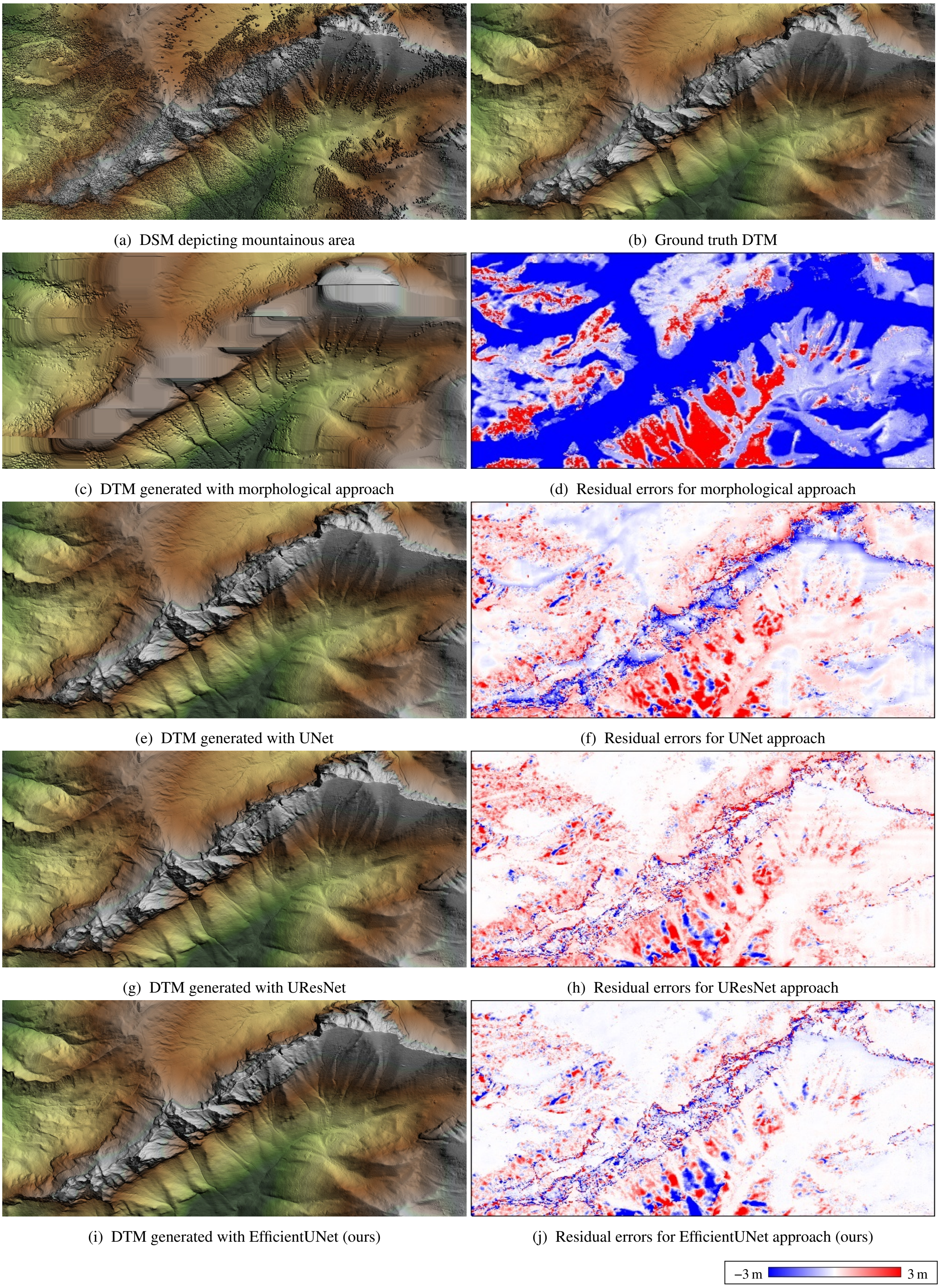
(i) EfficientUNet generated DTM (ours)

(j) Residual errors for EfficientUNet approach (ours)

Figure 3. Detailed visual analysis of DTMs, generated by (e) UNet , (g) UResNet, (c) morphology based methodology Krauß et al. (2008) and (i) the proposed EfficientUNet out of (a) initial DSM in comparison to (b) referenced terrain model over selected sample area depicting urban environment in Fribourg city. The images are color-shaded for better visualization. Additionally, residual error maps between generated DTMs with respect to the ground-truth are illustrated on (d), (f), (h) and (j).

the purpose to better demonstrate the strength of the proposed methodology.

In Figure 3 we compare for an urban environment the performances of the classical morphology approach, two commonly used network architectures and our presented approach. The selected sample scene has not only densely placed houses within a city but also very dense forest (see Figure 3a). By investigating the obtained DTMs, one can say that all methods succeeded to detect and remove above-ground information, however with a different level of accuracy. The morphology method performed the worst. The city area is just smoothed out and obviously changed, without preserving the areas which do not have above-ground information. Also the inhomogeneity in the forest environment exist in form of holes. Moreover, no terrain informa-



(a) DSM depicting mountainous area

(b) Ground truth DTM

(c) DTM generated with morphological approach

(d) Residual errors for morphological approach

(e) DTM generated with UNet

(f) Residual errors for UNet approach

(g) DTM generated with UResNet

(h) Residual errors for UResNet approach

(i) DTM generated with EfficientUNet (ours)

(j) Residual errors for EfficientUNet approach (ours)

Figure 4. Detailed visual analysis of DTMs, generated by (c) morphology based methodology Krauß et al. (2008), (e) UNet, (g) UResNet, and (i) the proposed EfficientUNet out of (a) initial DSM in comparison to (b) referenced terrain model over selected sample area depicting mountain landscape in Fribourg city. The images are color-shaded for better visualization. Additionally, residual error maps between generated DTMs with respect to the ground-truth are illustrated on (d), (f), (h) and (j).

Method	Error			
	MAE	RMSE	MedErr	NMAD
Morphology	3.522	8.026	-0.624	3.201
UNet	0.419	0.991	0.033	0.253
UResNet	0.354	0.860	**0.017**	**0.065**
EfficientUNet (ours)	**0.270**	**0.758**	-0.019	0.107

Table 1. Height difference statistics between generated DTMs and the reference DTM, evaluated over selected test area of Fribourg city. All values are meters.

Method	Error			
	MAE	RMSE	MedErr	NMAD
Morphology	1.327	2.215	-0.465	1.752
UNet	0.441	0.891	0.111	0.160
UResNet	0.363	0.767	0.046	**0.120**
EfficientUNet (ours)	**0.276**	**0.621**	**-0.009**	0.133

Table 2. Height difference statistics between generated DTMs and the reference DTM, evaluated over selected test urban area of Fribourg city depicted in Figure 3. All values are meters.

Method	Error			
	MAE	RMSE	MedErr	NMAD
Morphology	14.247	29.057	-2.209	9.699
UNet	0.824	1.826	0.142	0.648
UResNet	0.652	1.642	0.135	0.318
EfficientUNet (ours)	**0.559**	**1.627**	**-0.001**	**0.272**

Table 3. Height difference statistics between generated DTMs and the reference DTM, evaluated over selected test mountain area of Fribourg city depicted in Figure 4. All values are meters.

tion, such as hills or lowlands, is preserved. That can be clearly seen on Figure 3d depicting residual errors.

Investigating deep learning based approaches on Figures 3e, 3g and 3i, it is noticeable that they can better adapt to terrain profiles and actually keep them. All networks were able to detect above-ground objects, like trees and buildings, and filter them out. Roads, hills and lowlands are still visible and their features were not changed much. But further examining the results, some differences between the achievement of the three architectures can be observed. For example, the forest area on the right was the best removed by EfficientUNet. The resulted surface is less rough and has more precise heights, in comparison to UNet and UResNet generated surfaces in this area. Analysing the residual error map of the urban environment on Figure 3h, one can notice that the UNet generated surface has many areas, where ground was set to much lower heights (blue coloured areas), but the rest of the area is higher than original heights (red coloured areas on the ground which should not be changed). Investigating the UResNet residual map on Figure 3h, it can be observed that the above-ground objects are quite strongly visible on the resulted DTM. The possible explanation can be that this type of network is a good detector and it manages to identify all objects very precisely. However, it cannot assign a very correct height to the areas, where objects were removed. Analysing both the surface area Figure 3i and the residual map Figure 4j, we can say that EfficientUNet model performed the best on this area. It looks very similar to the ground truth DTM: all necessary features, such as lines and edges are preserved, hills and lowlands are unchanged, the forest is completely removed. The differences to the referenced DTM are minimal.

Overall, one can say that with an appropriate deep neural network architecture one can generate much better DTMs within urban areas with a mixture of small and very large buildings which was very problematic to do in the past.

We go on by testing the limits of investigated approaches on more challenging area – mountains. The selected scene of a complex relief together with resulted and reference DTMs are depicted on Figure 4. Starting with investigation of morphology-based generated DTM shown on Figure 4c, we can immediately conclude that the methodology fails on such places since it strongly over-smoothed any steep reliefs by flatting peaks and sharp edges. The resulted DTM does not have a realistic appearance in comparison to reference and deep learning DTMs. Furthermore, the terrain information is again very roughly estimated within a vegetated area and has many outliers. This investigations are even strongly observable on difference map (see Figure 4d). On the other hand, deep learning approaches were able to preserve all contours of complex mountain relief and on the first sign they look very similar to each other. However, investigating residual error maps, the differences can be found and they are correlated with the performance on urban areas. Analysing Figure 4f, it is obvious that UNet again performed the worst in producing the correct height within the whole area. Peaks of mountain relief are lower. The vegetation on the slope areas was not completely removed. On the other hand, Figure 4h demonstrates that UResNet is able to keep the areas without above-ground information unchanged, including the complex mountain peak region, much better in comparison to UNet. However, the areas, where the initial DSM was covered with vegetation, were not correctly reconstructed and are still higher than the referenced DTM. The resulted DTM from the EfficientUNet resembles the appearance of the referenced terrain the most, which is supported by the residual map on Figure 4j. The surface is smooth, less hint of vegetation is left on the resulted terrain. The steep relief is very well preserved and features all terrain details from the initial DSM. Only minor differences between the referenced DTM and ours are observable. This experiment again supports the fact that EfficientUNet model outperforms the rest of analysed methodologies.

To quantify the generated DTMs, we evaluated the proposed metrics for the selected test area depicted on Figure 1 on all setups, and their performances are reported in Section 5. In general, a significant difference can be observed in all metrics between deep learning approaches and morphology-based one. It is reasonable, since morphology-based approach was not able to keep important features, like mountain peaks, steep terrain relief or urban contours, and just flatted them out in most cases. On the contrary, deep learning approaches better adapted to such terrain profiles that is totally supported by all evaluated metrics.

We go on by studying the differences in the performance between presented deep learning architectures. Lower numbers of MAE and RMSE metrics for DTM generated with EfficientUNet in comparison to UNet and UResNet only further support our qualitative investigation. This is reasonable, since each of these both models have some problems on generated terrain. As the MedErr shows only a 1.9 cm difference for EfficientUNet, the large difference between RMSE and NMAD indicates that the remaining errors are not normally distributed, and visual inspection shows the few remaining differences concentrate on steep terrain boundaries and forests, which are difficult to filter out without ground points, particularly in mountainous terrain.

We additionally performed the same quantitative evaluation on two selected regions, urban and mountain, shown on Figure 3 and Figure 4, respectively, to investigate if different landscape characteristics influence the performance of the model. Both Section 5 and Section 5 have only further confirmed the superiority of EfficientUNet over UNet and UResNet, and its independence from the terrain relief.

6. CONCLUSION

In this work, we present a methodology that generates digital terrain models (DTMs) from digital surface models (DSMs) with the help of efficient deep learning neural network. The proposed EfficientUNet framework is based on an EfficientNet architecture used as encoder which is smaller than existing popular baselines. The approach is end-to-end and useful to automatically recognize non-ground pixels on a DSM and estimate the bare-ground height information in a way, that there are no differences in the heights between the surrounding bare-ground area and the resulting ones. EfficientUNet is particularly good at keeping a very complex relief of landscapes. It is able to reconstruct DTMs at a large-scale and, in our experiments, achieved lower MAE, RMSE and NMAD in comparison to the traditional morphology approach, and other deep learning baselines. Future research includes testing the model on different locations to explore its geographical generalization, comparing it against another non-deep learning algorithms, training a model on the combination of data from different sensors, such as light detection and ranging (LiDAR)-derived images, photogrammetric DSMs from both aerial and satellite images with different ground sampling distances (GSDs) to set up one generic model for different scenarios.

References

Amirkolaee, H. A., Arefi, H., Ahmadlou, M., Raikwar, V., 2022. DTM extraction from DSM using a multi-scale DTM fusion strategy based on deep learning. *Remote Sensing of Environment*, 274, 113014.

Arefi, H., Engels, J., Hahn, M., Mayer, H., 2007. Automatic DTM generation from laser-scanning data in residential hilly area. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36.

Bittner, K., Liebel, L., Körner, M., Reinartz, P., 2020. Long-short skip connections in deep neural networks for DSM refinement. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 43(B2), 383–390.

Duan, L., Desbrun, M., Giraud, A., Trastour, F., Laurore, L., 2019. Large-scale DTM generation from satellite data. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 0–0.

Gevaert, C., Persello, C., Nex, F., Vosselman, G., 2018. A deep learning approach to DTM extraction from imagery using rule-based training labels. *ISPRS journal of photogrammetry and remote sensing*, 142, 106–123.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hu, J., Shen, L., Sun, G., 2018. Squeeze-and-excitation networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7132–7141.

Huang, Y., Cheng, Y., Bapna, A., Firat, O., Chen, D., Chen, M., Lee, H., Ngiam, J., Le, Q. V., Wu, Y. et al., 2019. Gpipe: Efficient training of giant neural networks using pipeline parallelism. *Advances in neural information processing systems*, 32.

Hui, Z., Hu, Y., Yevenyo, Y. Z., Yu, X., 2016. An improved morphological algorithm for filtering airborne LiDAR point cloud based on multi-level kriging interpolation. *Remote Sensing*, 8(1), 35.

Khosravipour, A., Skidmore, A. K., Isenburg, M., 2016. Generating spike-free digital surface models using LiDAR raw point clouds: A new approach for forestry applications. *International journal of applied earth observation and geoinformation*, 52, 104–114.

Kingma, D. P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Krauß, T., Arefi, H., Reinartz, P., 2011. Evaluation of selected methods for extracting digital terrain models from satellite born digital surface models in urban areas. *Proceedings of the International Conference on Sensors and Models in Photogrammetry and Remote Sensing*, 5, 1–7.

Krauß, T., Lehner, M., Reinartz, P., 2008. Generation of coarse 3D models of urban areas from high resolution stereo satellite images. *International Archives of Photogrammetry and Remote Sensing*, 37(B1), 1091–1098.

Krauß, T., Reinartz, P., 2010. Urban object detection using a fusion approach of dense urban digital surface models and vhr optical satellite stereo data. *Proceedings of the ISPRS Istanbul Workshop*, 1–6.

Marmanis, D., Adam, F., Datcu, M., Esch, T., Stilla, U., 2015. Deep neural networks for above-ground detection in very high spatial resolution digital elevation models. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(3), 103.

Pijl, A., Bailly, J.-S., Feurer, D., El Maaoui, M. A., Boussema, M. R., Tarolli, P., 2020. TERRA: Terrain extraction from elevation rasters through repetitive anisotropic filtering. *International Journal of Applied Earth Observation and Geoinformation*, 84, 101977.

Pingel, T. J., Clarke, K. C., McBride, W. A., 2013. An improved simple morphological filter for the terrain classification of airborne LIDAR data. *ISPRS journal of photogrammetry and remote sensing*, 77, 21–30.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical image computing and computer-assisted intervention*, Springer, 234–241.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.-C., 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4510–4520.

Schuegraf, P., Schnell, J., Henry, C., Bittner, K., 2022. Building section instance segmentation with combined classical and deep learning methods. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2, 407–414.

Sithole, G., Vosselman, G., 2001. Filtering of laser altimetry data using a slope adaptive filter. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34(3/W4), 203–210.

Stucker, C., Schindler, K., 2020. Resdepth: Learned residual stereo reconstruction. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 184–185.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–9.

Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., Le, Q. V., 2019. Mnasnet: Platform-aware neural architecture search for mobile. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2820–2828.

Tan, M., Le, Q., 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. *International conference on machine learning*, PMLR, 6105–6114.

Tapper, G., 2016. Extraction of DTM from satellite images using neural networks. Master's thesis, Linköping University, Computer Vision.

Vosselman, G., 2000. Slope based filtering of laser altimetry data. *International Archives of photogrammetry and remote sensing*, 33(B3/2; PART 3), 935–942.

Wang, C.-K., Tseng, Y.-H., 2010. DEM generation from airborne lidar data by an adaptive dual-directional slope filter. *Proceedings of ISPRS TC VII Symposium—100 Years ISPRS*.

Wang, J., Shan, J., 2009. Segmentation of lidar point clouds for building extraction. *American Society for Photogrammetry and Remote Sensing Annual Conference*, 9–13.

Weidner, U., Förstner, W., 1995. Towards automatic building extraction from high-resolution digital elevation models. *ISPRS journal of Photogrammetry and Remote Sensing*, 50(4), 38–49.