remote sensing

MDPI

Article

IM2ELEVATION: Building Height Estimation from Single-View Aerial Imagery

Chao-Jung Liu [1], Vladimir A. Krylov [2,*], Paul Kane [3], Geraldine Kavanagh [3] and Rozenn Dahyot [1]

1 School of Computer Science and Statistics, Trinity College Dublin, Dublin 2, Ireland; chliu@tcd.ie (C.-J.L.); rozenn.dahyot@tcd.ie (R.D.)
2 School of Mathematical Sciences, Dublin City University, Dublin 9, Ireland
3 Ordnance Survey Ireland, Dublin 8, Ireland; paul.kane@osi.ie (P.K.); geraldine.kavanagh@osi.ie (G.K.)
* Correspondence: vladimir.krylov@dcu.ie

Received: 24 July 2020; Accepted: 19 August 2020; Published: 22 August 2020

Abstract: Estimation of the Digital Surface Model (DSM) and building heights from single-view aerial imagery is a challenging inherently ill-posed problem that we address in this paper by resorting to machine learning. We propose an end-to-end trainable convolutional-deconvolutional deep neural network architecture that enables learning mapping from a single aerial imagery to a DSM for analysis of urban scenes. We perform multisensor fusion of aerial optical and aerial light detection and ranging (Lidar) data to prepare the training data for our pipeline. The dataset quality is key to successful estimation performance. Typically, a substantial amount of misregistration artifacts are present due to georeferencing/projection errors, sensor calibration inaccuracies, and scene changes between acquisitions. To overcome these issues, we propose a registration procedure to improve Lidar and optical data alignment that relies on Mutual Information, followed by Hough transform-based validation step to adjust misregistered image patches. We validate our building height estimation model on a high-resolution dataset captured over central Dublin, Ireland: Lidar point cloud of 2015 and optical aerial images from 2017. These data allow us to validate the proposed registration procedure and perform 3D model reconstruction from single-view aerial imagery. We also report state-of-the-art performance of our proposed architecture on several popular DSM estimation datasets.

Keywords: building height estimation; digital surface model; optical aerial imagery; aerial Lidar; image coregistration; convolutional neural networks

1. Introduction

High-resolution orthorectified imagery acquired by aerial or satellite sensors is well known to be a rich source of information with high geolocation accuracy. These images are widely used in geographic information systems (GIS), for instance, for detection of man-made objects (building), urban monitoring, and planning. However, these optical images do not contain height information, and therefore this limits the scope of analysis that can be done based on aerial optical capture. Point clouds, on the other hand, provide complementary 3D information. Stereo image pairs [1], structure from motion (SfM) [2], or Light Detection and Ranging (Lidar) laser-scanning technology are traditionally used to obtain point clouds. These methods provide 3D information with various levels of accuracy which can then be converted to Digital Surface Model (DSM) which in turn can be stored as grayscale imagery. The latter can be used directly to annotate the height value on the aerial image. However, these methods require high computational resources and laborious fine-tuning. Moreover, using stereo image pairs with SfM requires image matching, which helps to estimate camera poses with different

temporal intervals. The height information is extracted via triangulation from pairs of consecutive views, and therefore the single view imagery can not be used by these techniques.

In this work, we focus on the scenario where building heights information is to be extracted in a fully automated mode from a single airborne or satellite optical image without relying on the availability of any further contemporary or historical imagery, point clouds, or GIS records data. To arrive at this we propose a convolutional neural network (CNN) architecture IM2ELEVATION that takes a single optical image as input and produces an estimated DSM image as output, see Figure 1. Our architecture is inspired by [3,4] and features additional skip connections and postprocessing convolutional layers which deliver highest performance on the task at hand. To enable the training process, we use a set of point cloud data that serves as ground truth reference. In particular, we investigate the use of publicly available point cloud data of Dublin city center recorded in 2015 [5] for extracting the training building height information. This is used in combination with the more recent aerial imagery collected by Ordnance Survey Ireland (OSI) in 2017. These data are of higher resolution than other alternative open source datasets, see, e.g., in [6]. We also explore co-registration required to perform inference on these multisensor data, which is needed because of the misalignment in space and time between the streams. Figure 2 presents the Dublin dataset used for this study and visualizes the height estimation process.

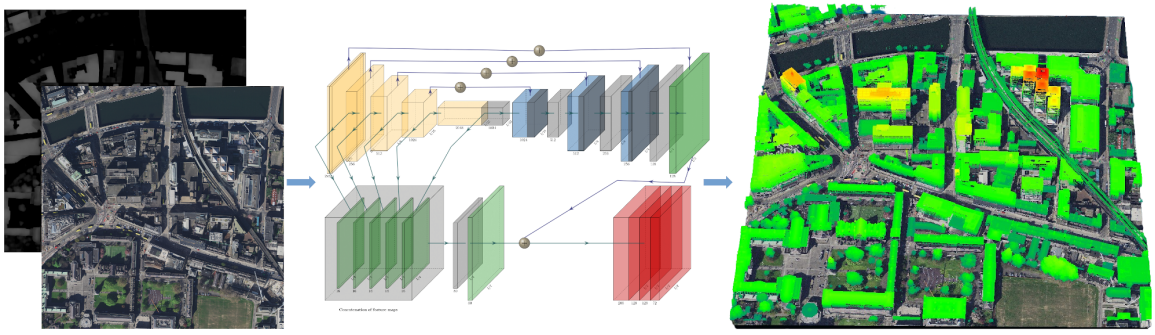
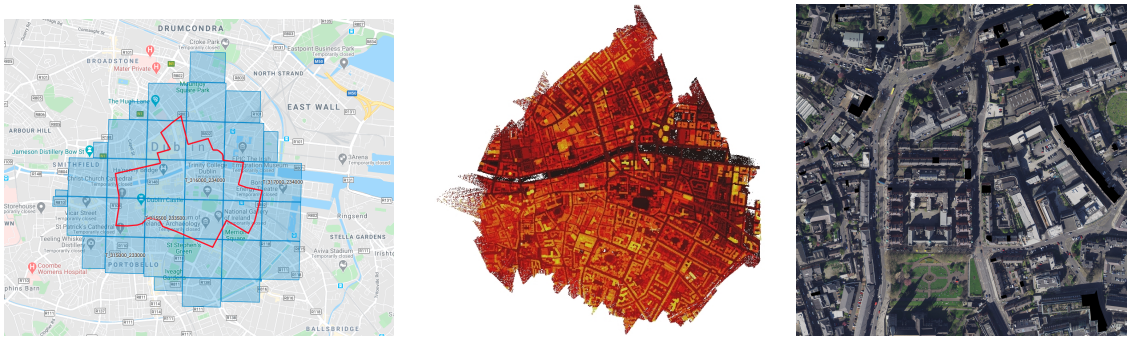


Figure 1. IM2ELEVATION pipeline: from single view aerial imagery to building heights/Digital Surface Model (DSM). Light detection and ranging (Lidar) point cloud is used solely at training phase.



aerial tiles (blue) and Lidar cover (red) 2015 Lidar point cloud dataset sample aerial image tile

Figure 2. Study case of building heights in Dublin (Ireland).

Using supervised learning approach for converting automatically an aerial image (input) into a DSM image (output), our first step (see Section 3) is to create a training dataset of aligned pairs (aerial-DSM). Using the resulting training set, we then train a CNN for inferring building heights (see Section 4). Our approach is validated against state of the art benchmarks in Section 5, discussed in Section 6 and we conclude the paper in Section 7.

2. Related Works

Using multiple sources of data for inference has a number of applications in mapping (see Section 2.1) but presents challenges for machine learning techniques where the quality of the data used for training directly impacts the performance of the resulting techniques. Here, we focus on buildings and the corresponding roof shapes that are important for instance for solar panels [7] and 5G deployment [8] (see Section 2.2). To capture 3D data directly is more expensive in practice than capturing multispectral aerial imagery, therefore some research efforts have been deployed for extracting depth information from single view images (see Section 2.3) and extended for processing aerial images to infer heights (see Section 2.4). We also report a brief review of the registration techniques applied to point clouds and aerial imagery (Section 2.5) as this is inevitably the first step in dealing with incoming new data, see, e.g., Figure 2.

2.1. Fusion of Heterogeneous Data Streams

In recent years, various sources of multimedia input and GIS data have been used in numerous applications such as mapping and 3D city reconstruction. For instance, social media data and satellite imagery used together provide an opportunity to measure and monitor the impact of flooding [9]. Social media data has the advantage to provide up-to-date information capturing geolocated information and sentiment [10]. Images posted on social media is however too sparse and noisy for applications such as 3D city reconstruction. Street-level imagery (e.g., Google Street View) has been used for 3D reconstruction [11], for object geolocation such as poles for asset management [12], and trees for monitoring biodiversity [13]. Street-level imagery, when used with building shape information extracted from OpenStreetMap (OSM), also provides opportunities to generate light 3D model of cities usable with game engine technologies [10]. Extending the point of view from street level to aerial, Liu et al. [14] proposed to use OSM to segment and propagated labels to a point cloud from drone imagery collected over Trinity College Dublin campus in 2017 [15].

2.2. Mapping of Buildings and Rooftops

Detection and segmentation of buildings in satellite and aerial imagery are standard processes for populating and efficient updating of modern maps [16–18]. For instance, Lafarge et al. [17] presented an approach for building reconstruction from a single Digital Surface Model (DSM) with buildings treated as an assemblage of simple urban structures extracted from a library of 3D parametric blocks. As an alternative to DSM, Benedek et al. [16] introduced a probabilistic method which integrates building extraction with change detection in remotely sensed image pairs.

Beyond building footprint segmentation in 2D and 2.5D (DSM) aerial imagery, knowledge of roof geometry is also essential for instance in detecting and assessing the potential of rooftop solar photovoltaic (PV) installations on local electricity networks for sustainable development. Palmer et al. [19] proposed an approach for roof PV suitability based on slope and aspect using aircraft-based Lidar data, building footprint data, GIS tools, and aerial photography. Song et al. [20] proposed to extract the 2D rooftop outlines and 3D rooftop parameters retrieved from high-resolution remote sensing image data and DSM considering five rooftop categories (flat rooftops, shed rooftops, hipped rooftops, gable rooftops, and mansard rooftops).

2.3. Monocular Depth Estimation from Images

While the stereo vision has been intensely researched to use triangulation cues from two or multiple images, it cannot be applied to the case where only a single image is available. Instead, there are numerous monocular cues such as texture variations and pixel's gradient, occlusion, and color/haze, which can be used to infer the underlying 3D structure. Single-view depth estimation can be classified into two major groups, specifically random field-based approaches and deep convolutional neural network (DCNN) approaches. Saxena et al. [21,22] segmented an image into small patches

(super-pixel) and Markov Random Fields were applied to infer depth for each of the segmented patches. Their model is trained based on several features at different scales [21,22]. A similar approach by Wei et al. [23] introduced hierarchical representation of the scenes with depth prediction model based on Conditional Random Fields (CRF). The latter encoded interactions within and across different layers, jointly exploiting local and global 3D information. More recently DCNN-based models have been actively investigated [24–29]. Eigen et al. [30] proposed to use two pipelines of DCNNs to extract features at different scale levels: one that regressed global depth structure from a single image, and another for capturing the high level features. Both outputs were concatenated and are collectively refined in the final layer. This has later been extended to handle multiple tasks like semantic segmentation and surface normal estimation [24].

Laina et al. [28] and Xu et al. [26] demonstrated the use of CRFs with DCNN-based models. These hybrid DCNNs methods capitalize on exploring the strength on pairwise pixel interaction whereas the DCNNs only consider the unary potentials during training. In the similar fashion, Long et al. [27] leveraged connected CRF as part of the fully convolution network layer to jointly maximize the posterior on predicting semantic labels. The encoder-decoder framework has been a common trend in DCNN architecture [3,29,31–33]. The encoder transforms the imagery into a latent space which contain high level features of the imagery. The decoder is to increase the spatial resolution of the feature maps generated by the encoder. This enables the networks to regress the input to desired output. Laina et al. [28] used the ResNet-50 [32] as the DCNN architecture's backbone and replaced the fully connected layer to the up-projection blocks which act like reverse operation of pooling, scaling up the spatial resolution to half the size of input. Mal et al. [29] used up-projection blocks [28] in deconvolutional layers to learn mapping from a sparse depth map to a dense depth map. Both [28,29] used similar encoder–decoder fashion architectures to predict depth information from single image, however their results are underperforming in terms of preserving the object shapes.

Hu et al. [4] combined the up-projection blocks with the concept of CNN skip connections [34]. In addition, the features from encoder are fused into the same size of block as multifeature fusion block (MFF). The MFF is then concatenated to the output of decoder. Furthermore, they use gradients and normals as part of the loss function so that the designed network is be more aware of the edges and shapes during training.

2.4. Aerial Image Height Estimation

Unlike monocular depth estimation in computer vision, there has been relatively little on estimating depth from a single aerial image. The methods that are used in monocular depth prediction can be considered as the problem of estimating distance between sensor to the observed scene in remote sensing. Height estimation from single view is an ill-posed problem: a single 2D image may have infinite number of possible 3D scenes. The ambiguity lies in mapping from 3-channel RGB into a channel height value. Fully convolutional-deconvolutional network architectures are a useful tool that is capable of guiding the model through the process of learning this ambiguous mapping with considerable accuracy. Alidoost et al. [35] proposed to regress RGB image to DSM, together with linear line structure of roof in three different classes: eave, ridge, and hip lines. The lines were used as additional information in their building reconstruction process.

Architectures with skip connection are capable of keeping the global edge features and have been recently used for buildings height estimation [36,37]. Similar architectures were used to jointly estimate height and semantic labels [38,39] in order to improve accuracy by using the training information for these two complimentary tasks. While the skip connection has been widely used to directly concatenate the same spatial resolution features from encoder to decoder, we establish the superior performance of multiscale feature fusion from encoder blocks (see Section 4).

Ghamisi et al. [40] exploited conditional Generative Adversarial Network (cGAN) to generate a synthesized DSM from a single aerial image. Bittner et al. [41] used cGAN to refine the building roof surface by regressing DSM from stereo aerial image to level of detail (LoD) 2 building shape.

2.5. Point Cloud and Aerial Imagery Registration

Here, we will briefly review relevant methods for coregistration of the specific type that is required in our work: point clouds and optical aerial imagery.

In feature-based registration methods, different types of features, such as points, linear features, and patches, are extracted to achieve the correspondence between modalities. Habib et al. [42] extracted features of lines and patches. Kwak et al. [43] used centroid points of the roofs. Peng et al. [44] used lines and intersections (corners). Zhang et al. [45] used the objects from point cloud to form a geometry constraint (e.g., polygon). An assumption was made that all points in point cloud within the geometry constraint in 2.5D fall into the same corresponding object in the optical imagery. The constraint was used to find the optimal registration solution. This method is similar to Liu et al. [14], which registered polygon from OSM to 3D point cloud data. A different method from Chen et al. [46] employed deep learning method for training registration between polygons and 2D aerial images. The network learned the transformation so as to transform polygons to correct positions. However, feature-based methods rely on physical correspondences, which are not always available, especially when using datasets that have been acquired at substantially different time points.

Information theory has been used extensively for multimodal registration. The statistical similarity can be measured by Mutual Information (MI), which utilizes the statistical dependencies between the data to be registered and derives similarity measurements [47]. Using MI, Mastin et al. [48] proposed to optimize the rendering from point cloud via tuning extrinsic camera parameters. Parmehr et al. [49] registered aerial imagery onto LiDAR point clouds by maximizing the combined mutual information from the grayscale-encoded height, return-pulse intensity, and optical imagery.

3. Data Preprocessing and Registration

We outline the employed data preprocessing for Lidar data in Section 3.1 and for aerial imagery in Section 3.2. The registration procedure we propose to align our two heterogeneous and asynchronous datasets (aerial images 2017 and Lidar point cloud 2015) is presented in Section 3.3, and then validated to improve robustness (see Section 3.4). The preprocessing allows us to create a dataset for training and testing our CNN model that converts aerial images (input) into DSM (output).

3.1. Preprocessing of Lidar Data

Our purpose here is to generate DSM ground truth from the available Lidar point cloud. These DSM images are then used as the reference for training and testing our CNN.

In our case study, we work on a 2015-captured Lidar dataset that covers the city center of Dublin, with ~5.6 km^2 including partially covered areas [5]. The flight altitude was ~300 m and there are 41 flight path strips in total. The final Lidar point cloud was generated from the registration of points from these strips. As this Lidar point cloud contains more than 1.4 billion points, it is split into 0.25 km^2 tiles to be loaded and processed efficiently. The point cloud density inside the projected area varies from 238 to 348 points/m^2. There is ~3.5 km^2 complete point cloud coverage, which is equivalent to 14 tiles. The average accuracy of this point cloud is 0.03 m with a maximum of 0.3 m deviation. Tiles are back-projected to DSM (output of our neural network). The ground control sample distance (GSD) of the generated DSM is scaled to 15 ground resolution which is the resolution of the aerial images employed. The generated DSM from Lidar point cloud has holes caused by low density of points in certain areas. To enhance the DSM completeness, we fill those holes by interpolating from the closest available points. In addition, the DSM is saved in 16-bit image format which preserves more details about local gradient (see in Figure 3).

The return-pulse intensity is also obtained from the Lidar point cloud. It is merged with DSM (Figure 4) as an additional source of information for validating our registration process. Our unit voxel volume is set to 15 cm $\times$ 15 cm $\times$ 15 cm to be at the same resolution as the aerial imagery. The density

of the points in the Lidar point cloud is observed to be higher on vertical walls that translate as edges in aerial imagery and DSMs.

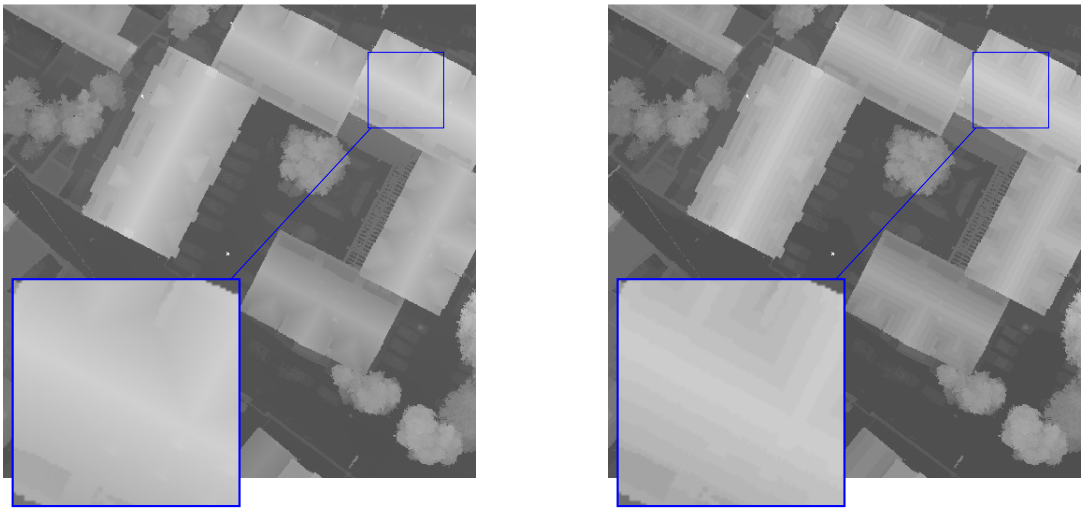


Figure 3. The roof has smoother gradient in 16-bit image (**left**) compared to the 8-bit one (**right**). Eight-bit DSM can only capture details up to 1 m resolution, whereas 16-bit DSM is capable of capturing at 0.15 m resolution.

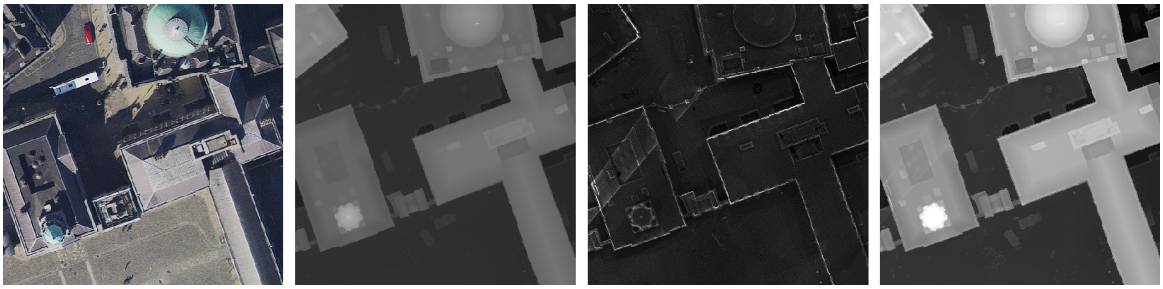


Figure 4. From **left** to **right**: optical imagery, DSM, return-pulse intensity, and fusion of DSM and return-pulse intensity. The optical image and the DSM are used in Mutual Information (MI) registration process whereas the fusion and the optical image are used in Hough validation. Both DSM and return-pulse intensity are derived from the point cloud source

3.2. Preprocessing of Aerial Orthorectified Imagery

The aerial image we use in our study is captured by OSI in 2017 and covers the entire extent of the Lidar 2015 dataset at 15 cm geometric resolution. Note that the Lidar dataset 2015 [5] also provides aerial imagery, however it only covers partially Dublin city (see Figure 2). Therefore, we deal with two distinct datasets and there are two major challenges: (1) The change of scenes between the two captures (in 2015 and in 2017), and (2) the two data streams are not properly aligned with each other even though both of them are georeferenced. Small deviations can be observed that originate from different projection methods and sensor calibration errors.

We first remove the areas that change substantially between the acquisition dates of the two datasets, this mainly occurs on dynamic objects, e.g., cars and buses. Only areas with DSM values above 2.5 m from the ground are kept to mitigate such discrepancies. We then convert RGB optical images to grey scale so that they can be used to perform registration with DSM via Mutual information registration (explained Section 3.3). Return-pulse intensity and DSM images are merged (see Figure 4) and then used with the optical image for validation of our MI registration quality (explained Section 3.4). This approach is chosen because DSM has a more similar distribution of intensities to optical imagery than return-pulse intensity, which leads to a better performance in MI registration. Moreover, fusing the

DSM and the return-pulse intensity retains more distinctive features alongside edges which boosts the Hough line extractor's performance.

3.3. Registration with Mutual Information

Mutual information is employed to carry out registration between aerial imagery and DSM. MI [49] measures the statistical similarity between datasets. The matching similarity is based on the intensity of distribution within images. The definition is as follows,

$$MI(I^P, I^D) = H(I^P) + H(I^D) - H(I^P, I^D) \tag{1}$$

$$H(I^P) = -\sum_{i=1}^{N} p_1^i \cdot \log(p_1^i)\ ,\ H(I^D) = -\sum_{i=1}^{N} p_2^i \cdot \log(p_2^i) \tag{2}$$

$$H(I^P, I^D) = -\sum p(i,j) \cdot \log(p(i,j)) \tag{3}$$

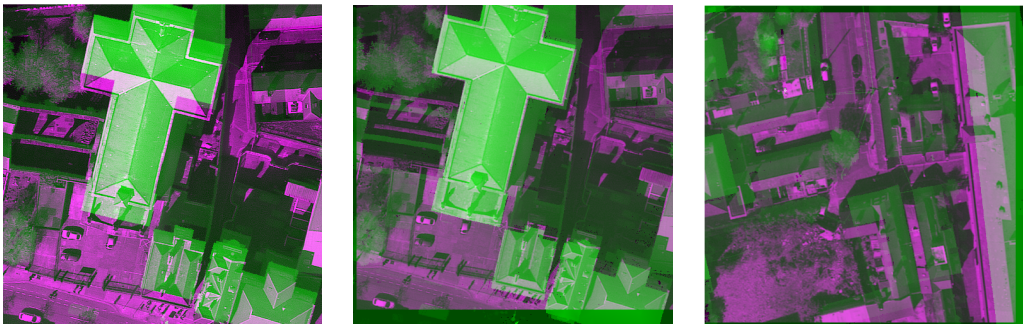
where $H(I)$ is the marginal (Shannon) entropy of image I [50], N is the number of bins used for the histograms of the images (estimated distribution p), and $H(I^P, I^D)$ is the joint entropy.

The aerial image I^P is chosen as the reference source on which the DSM image I^D is aligned. The statistical dependence between I^P and I^D is computed using probability density function (pdf) estimated via histogramming using a non-parametric approach. One-plus-one evolutionary algorithm [51] was used in the optimizer. It perturbs the matrix parameters in different direction to find the maximum MI value. In the case of misregistration, the output from the joint pdf $p(i,j)$ yields large negative number, which increases the value of the joint entropy. After appropriate registration, the images are expected to yield the smallest joint entropy. A parametric registration transformation between the images is formulated as a translation operation. Disabling rotation and scaling operations is a necessary trade-off to enable efficient and robust registration as otherwise the process is highly affected by smaller objects other than buildings. The translation T is estimated as follows,

$$\hat{T} = \arg\max_{T} MI(I^P; I^D) = \arg\min H(I^P; I^D)\ ,\ \hat{T} \in R^2. \tag{4}$$

We ran 1000 iterations to get optimal solution $\hat{T}$. GPS location available with the images is used to initialize T. Note that the initial level of registration between the available Lidar and optical imagery, both of which are georeferenced, allows for only approximate matching and we resort to an iterative procedure to improve the quality of local matching. We denote the aligned DSM image as $\hat{I}^D$.

The registration process can fail, as demonstrated in Figure 5, if a scene has undergone substantial changes in between the acquisitions. To detect such situations, the Hough line extractor is applied to aerial images and the corresponding fusion of DSM and intensities (Section 3.4). We compute the distances between the Hough detected lines in the two modalities and validate matching pairs. Invalid examples were adjusted via its neighbors' mean vector. This is used as the objective function to ensure the quality of the registration samples. The aim is to create a good training dataset to train our network: quality of the training data is crucial for the high performance of the resulting neural network.



Initial misalignment Successful registration Failed registration

Figure 5. MI registration between return-pulse intensities (green) and optical image (purple) may fail when the scene content is significantly different between the two data sets.

3.4. Patch Adjustment via Hough Lines Validation

To enhance the quality of pairs $\{(I_k^P, \hat{I}_k^D)\}$ for training our deep neural network, we exploit the Hough space to retain the good pairs of images registered with MI. Given two images I^P (optical imagery) and I^Q (fusion of DSM and return-pulse intensity), two sets of Hough lines, noted $V^P = \{\rho_i^P, \theta_i^P\}_{i=1}^{N^P}$ and $V^Q = \{\rho_i^Q, \theta_i^Q\}_{i=1}^{N^Q}$ are computed. The distance between parallel lines ($\theta^Q = \theta^P$) between I^P and I^Q is computed as

$$e = \|\rho^P - \rho^Q\| \tag{5}$$

Denoting e and e' the distance between the two lines before and after registration respectively, we validate a pair of patches having n corresponding lines when

$$\sum_{i=1}^{n}(e_i - e_i^{'}) \begin{cases} > 0, & \text{valid-registered pairs} \\ \leq 0, & \text{not valid-registered pairs} \end{cases}$$

Parallel lines were grouped together by searching the closest matching lines within a range $e < \delta$ before registration to reduce the impact of failed registration. To correct the detected cases of failed patch registration, the missing translation vectors are interpolated between the adjacent correctly registered neighbor patches (this is referred to as adjustment in the following).

4. CNN Network Design

We propose a deep learning architecture based on that presented in Hu et al. [4]. Specifically, we use a fully convolutional-deconvolutional network presented in Figure 6 that regresses a RGB (3 channel) image to a DSM image (1 channel). A Squeeze-and-Excitation Network [3] is used as encoder to extract features, generating five encoding blocks.

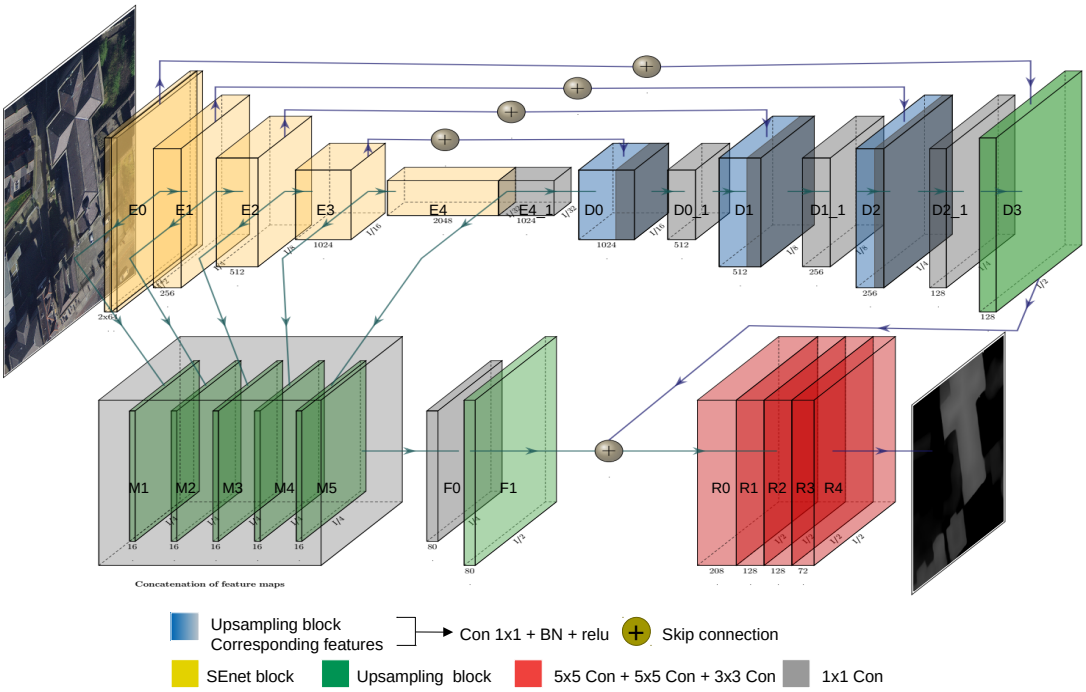


Figure 6. IM2ELEVATION architecture with extra skip connections and postprocessing layers. E: down-sampling block. D: up-sampling block. M: multi-fusion block. F: multi-fusion block processing. R: convolution layers.

The encoder (convolutional layers E0–E4) extracts features with resolution of $1/2$, $1/4$, $1/8$, $1/32$ of the original image. The decoder (deconvolutional layers D0–D3) upsamples the scale feature from $1/32$ to $1/2$ of the original image size. We use the decoder proposed by Laina et al. [28] with upsampled feature maps are concatenated with the corresponding features from the encoder, followed by a 1×1 convolution layers $D0\_1, D1\_1, D2\_1$, to reduce the number of channels. The latter blocks mix the weights and reduce the number of channels. Multiscale features (M1–M5) are extracted from encoder and resampled to the constant scale of $1/4$ of the initial image resolution, then mixed via 1×1 layer $F0$ and upsampled to the scale of $1/2$ of the input in $F1$. For visualization purposes, examples of several multiscale feature maps are presented in Figure 7. Finally, the last layer of deconvolution $D3$ is concatenated with $F1$ thus forming $R0$, and fed into convolution layers $R1 - R4$ in order to refine the final prediction map. Each convolutional except $R4$ layer features batch normalization and rectified linear unit (ReLU) activation function. The details of the input/output channels and the feature size are reported in Table 1. Further details on standard architectural elements can be found in [4,28].

Our architecture employs a skip connection that concatenates features directly from the encoder block with those from the multiscale features block. We observe a significant empirical improvement in our results due to this strategy, see Section 5.2.

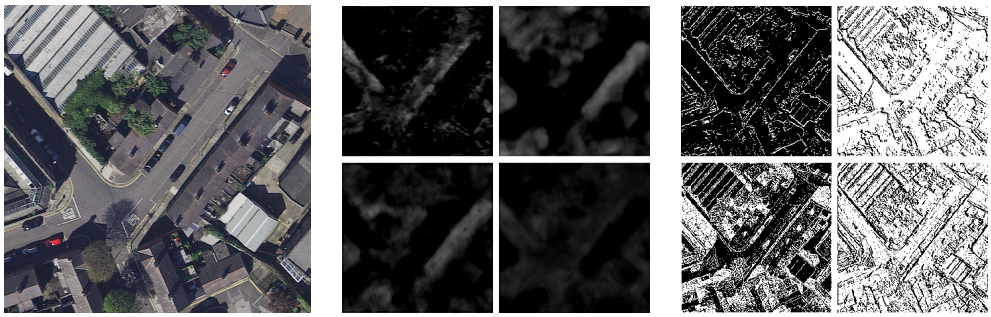


Figure 7. From **left** to **right**: aerial image, CNN features from Multi-feature block and CNN features from first layer of encoding block.

Table 1. IM2ELEVATION size of feature maps, and input/output channel based on SENet [3].

Layer	Output Size	Input/C	Output/C
E0	440 × 440	3	64
E0_1	220 × 220	64	128
E1	110 × 110	128	256
E2	55 × 55	256	512
E3	28 × 28	512	1024
E4	14 × 14	1024	2048
E4_1	14 × 14	2048	1024
M0	110 × 110	64	16
M1	110 × 110	256	16
M2	110 × 110	512	16
M3	110 × 110	1024	16
M4	110 × 110	2048	16
F0	110 × 110	80	80
F1	220 × 220	80	80
D0	28 × 28	2048	1024
D0_1	28 × 28	1024	512
D1	55 × 55	1024	512
D1_1	55 × 55	512	256
D2	110 × 110	512	256
D2_1	110 × 110	256	128
D3	220 × 220	128	208
R1	220 × 220	208	128
R2	220 × 220	128	128
R3	220 × 220	128	128
R4	220 × 220	72	1

The network loss function guides the training process to obtain the best fitting function modeling the training data. We construct the loss that consists of L_1 norm (Equation (7)), surface normal (Equation (9)) and spatial gradient (Equation (8)) as follows,

$$L = l_{depth} + \lambda\, l_{grad} + \mu\, l_{normal} \tag{6}$$

where $\lambda, \mu \in \mathbb{R}^+$ are weight coefficients. Here, we have

$$l_{depth} = \frac{1}{n}\sum_{n=1}^{i} F(e_i)\,, \quad e_i = \|\hat{h}_i - h_i\|_1, \quad F(x) = \ln(x+\alpha), \tag{7}$$

e_i is the L_1 norm between the estimated height $\hat{h}_i$ and ground truth height h_i. $\alpha > 0$ is a slack parameter to ensure $F(e_i) \in \mathbb{R}$ has a lower bound (in practice we always set $\alpha = 0.5$). In order to make the network more aware of the edge structure, we introduce l_{grad} defined as

$$l_{grad} = \frac{1}{n}\sum_{n=1}^{i}\left(F\left(\frac{\partial e_i}{\partial x}\right) + F\left(\frac{\partial e_i}{\partial y}\right)\right) \tag{8}$$

where (x, y) are the spatial coordinates in the residual image e.

l_{normal} is a cost on the normals to the surface of the estimated DSM with respect to its ground truth:

$$l_{normal} = \frac{1}{n}\sum_{n=1}^{i}\left(1 - \frac{\langle n_i^d, n_i^g\rangle}{\sqrt{\langle n_i^d, n_i^d\rangle}\sqrt{\langle n_i^g, n_i^g\rangle}}\right) \tag{9}$$

$\langle\cdot,\cdot\rangle$ denotes the inner product of vectors, which infers the orientation of surface. This cost penalizes the subtle changes of gradient which may not be captured in Equation (8). Besides, this guides the network to learn more geometric features.

5. Results

Our dataset consists of 14 tiles and each tile is split into 169 patches. Each patch is 500×500 pixels in size with 250 pixels overlapping for the purpose of increasing training data volume. Of the original 2366 patches, 1999 patches are used for training, and 367 patches are used for testing and for comparisons between several preprocessing pipeline scenarios: no registration (base), registration (MI), and registration with invalid patch adjustment (adjustment). In addition, we augment the training set by randomly flipping and jittering the training data. Adam solver is used with weight decay 0.001 and learning rate 0.0001. The learning rate is dropped by 10 percent in every 5 epochs. The CNN network is done with PyTorch package (The implementation will be made available upon publication at https://github.com/speed8928/IMELE.). Figure 8 demonstrates the evolution of the training/test loss throughout the training process: both MI and patch-adjusted datasets show steady loss decrease whereas the base dataset (without preprocessing) has large fluctuations (unstable) training loss. Similar behavior has been observed on the training/test Mean Absolute Error (MAE). We notice that beyond epoch 60 the test performance plateaus.

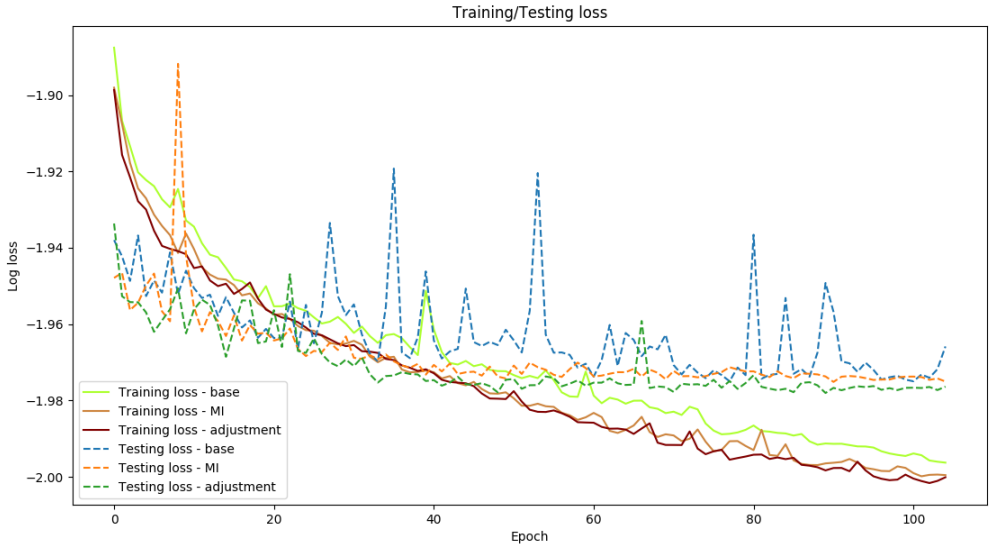


Figure 8. The training and testing loss on original data (base), MI-registered, and after Hough line patch adjustment. The registered data demonstrate more stable convergence than non-registered data.

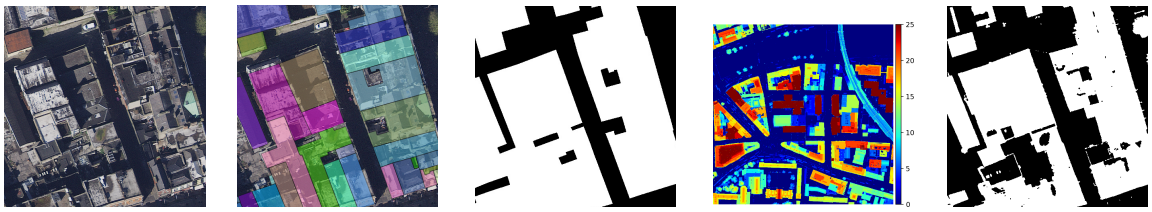
In Section 5.1 we assess the registration with Mutual Information. Section 5.2 presents our CNN improved performance when using registered and non-registered training sets.

5.1. MI Registration and Hough Line Validation

To evaluate the accuracy of registration, the vector layers for buildings provided by OSI are used as ground truth after being transformed to binary maps. This additional data source is used solely for the purpose of validating the performance of the registration pipeline and is not part of CNN training. For evaluating the performance of MI and the effect of validation with Hough transform, 11 tiles were selected. For each tile of image, 169 patches were generated, giving the total number of testing samples $K = 11 \times 169 = 1859$.

Figure 9 illustrates the data used in our experiments. For each test sample $k \in \{1, \cdots, K\}$, see Figure 9a, the total number of valid pixels associated with ground truth building label is denoted as n_g^k. The DSM is converted to a binary map for buildings by setting all pixel value to 0 if these have values inferior to 2.5 m above ground, and set to 1 otherwise (if superior to 2.5 m). The intersection defined by Equation (10) gives the average of the proportion of pixels from DSM building binary map (denoted M_h^k, see Figure 9e) corresponding with the ground truth building binary map (denoted M_g^k, see Figure 9c), normalized by n_g^k (the valid number of pixels from M_g^k):

$$\text{Intersection} = \frac{1}{K} \sum_{k=1}^{K} \frac{M_h^k \otimes M_g^k}{n_g^k} \tag{10}$$



(**a**) Sample k (**b**) building shape (**c**) ground truth M_g^k (**d**) DSM (**e**) binary DSM M_h^k

Figure 9. Example of data used for preprocessing and training.

Table 2 highlights the quantitative improvements associated with the use of MI and Hough validation preprocessing steps. Misregistered patches identified by Hough transform are adjusted by local translation that allows more meaningful computation of intersection for this evaluation (Method III). The value of MI after patch adjustment declines a little, whereas the value of Intersection increases. Further validation of preprocessing is reported in the next Section in conjunction with the proposed CNN.

Table 2. MI and Intersection obtained by applying registration (higher values are better). $n_g = 1.34$ bn. # patches = 1859. Note that we employ only a subset of training data due to the partial coverage of the building binary map. Best results in bold.

Methods	MI Registration	Data Adjustment (Hough Validation)	MI↑	Intersection↑
I	✗	✗	0.7650	0.8912
II	✓	✗	**0.9235**	0.9880
III	✓	✓	0.9193	**0.9926**

5.2. Height Inference with CNN

We now report the performance of the height inference CNN on the preprocessed Dublin dataset. The Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) are presented in Table 3. In

the comparison with the baseline architecture, Hu et al. [4] demonstrates that the proposed CNN improvements are of significance for the height estimation problem. We note that using registered data for training only improves the RMSE but deteriorates the MAE in comparison to non-registered training data. However, improvement on both can be found in registered data with patch adjustment.

Table 3. DSM estimation accuracy measured by MAE and RMSE (lower vaues are better): Hu et al. [4] vs. IM2ELEVATION (proposed) with different types of registration between Lidar and aerial data. The loss employs $\lambda = 1, \mu = 1$ for all methods. Best results in bold.

Method	Preprocessing	GSD	MAE (m)↓	RMSE (m)↓
Hu et al. [4]	none (I)	15 cm/pixel	1.99	5.04
	MI registration(II)		2.08	4.12
	MI, patch adjustment (III)		1.93	3.96
Hu et al. [4] + skip connection	MI, patch adjustment (III)	15 cm/pixel	2.40	4.59
IM2ELEVATION	MI, patch adjustment (III)		**1.46**	**3.05**

We tested our network on popular remote sensing datasets, including IEEE GRSS Data Fusion Contest dataset 2018 (DFC2018) [52,53] and ISPRS Potsdam and Vaihingen datasets [54,55], see Table 4. After retraining on these data, our network outperforms other the state-of-the-art methods [35,37,39,40] on DFC2018 and Potsdam datasets. We do not apply any preprocessing developed throughout Section 3 to these data to ensure comparability with benchmark methods. We employ ~25% randomly selected tiles for testing and the rest is used to train our pipeline for each of the 3 considered benchmark datasets. Note that in our work we employ solely the DSM information and make no use of semantic labels, which is an additional and expensive (effort-wise) source of complementary information. Nevertheless, our method still outperforms some of the other benchmark methods, which perform multitask learning. In particular, IM2ELEVATION outperforms other methods in MAE on DFC2018 dataset. Among the single task learning models, our model reaches the highest accuracy in RMSE in the Potsdam dataset. We also observe that the accuracy obtained with the Dublin dataset is in general at a comparable level which suggests that this dataset is reasonably balanced and our results can be generalized.

Table 4. State-of-the-art comparisons on popular datasets. Lower values of MAE and RMSE are better. * The authors of [39] perform multitask learning. ** The authors of [37] employ ZCA whitening on training data, so the results may not be comparable directly.

Methods	Training Input	GSD	MAE (m)↓	RMSE (m)↓
IEEE DFC2018 dataset				
Carvalho et al. [39]	DSM	5cm/pixel	1.47	3.05
Carvalho et al. [39] *	DSM + semantic		1.26	2.60
IM2ELEVATION, $\lambda = 1, \mu = 1$	DSM		1.19	2.88
ISPRS Vaihingen dataset				
Amirkolaee et al. [37] with ZCA whitening **	DSM	8cm/pixel	-	2.87
IM2ELEVATION, $\lambda = 0, \mu = 0$	DSM		2.96	4.66
ISPRS Potsdam dataset (nDSM)				
Amirkolaee et al. [37] with ZCA whitening **	DSM	8cm/pixel	-	3.46
Alidoost et al. [35]	DSM		-	3.57
Ghamisi et al. [40]	DSM		-	3.89
IM2ELEVATION, $\lambda = 0, \mu = 0$	DSM		1.52	2.64

We demonstrate the obtained DSM estimation results in a full tile in Figure 10. It can be readily observed that the building contours present high variance in general. This can partially be attributed to remaining artifacts after registration and orthorectification processes. Buildings substantially taller than average in the scene appear to be more challenging for monocular height estimation as well.

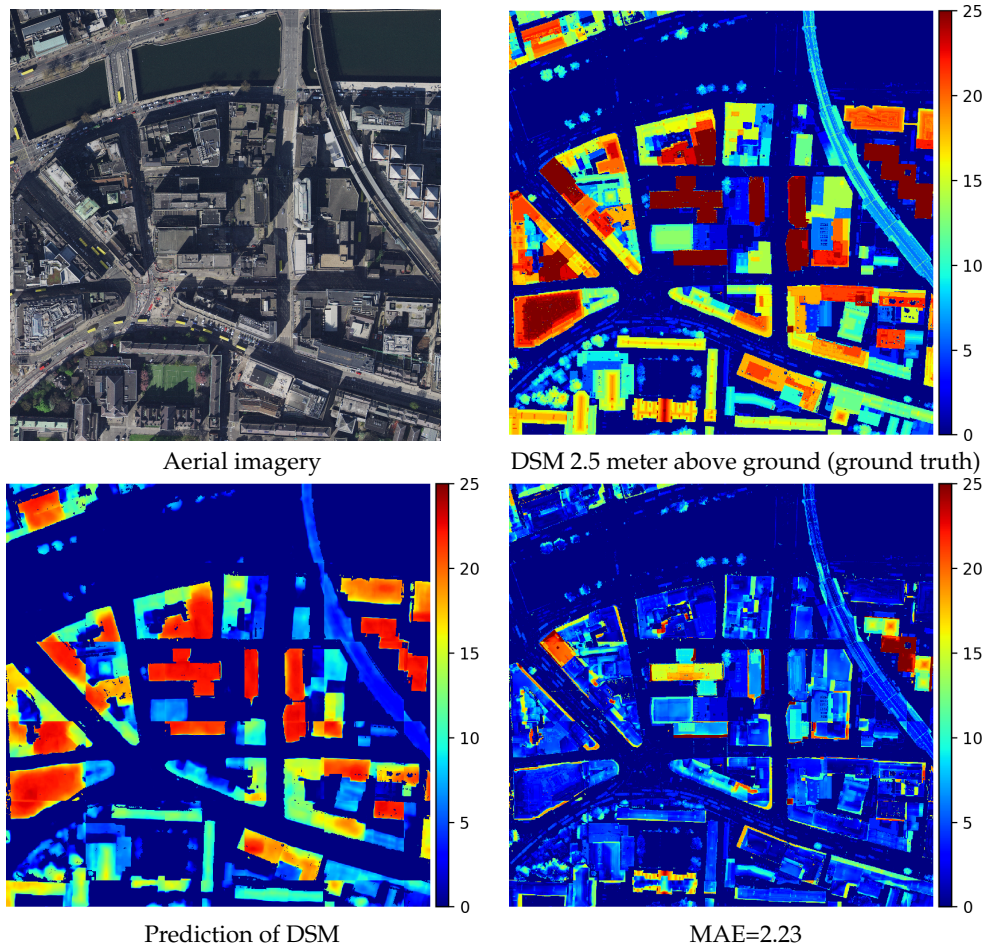


Figure 10. DSM of a 500 × 500 m tile in the Dublin inner city center with northing of 316,000 to 316,500, and easting from 234,000 to 234,500, in TM65 projection.

5.3. Height Inference from Stereoscopic Imagery

The height inference on stereo images is reported in this section to compare with our height inference performed on a monocular set of airborne imagery. In remote sensing, structure from motion (SFM) is often adapted to infer the elevation of the landscape [56]. The point cloud can be generated by rectifying a selected pair of images, followed by computing the disparity map which is similar to DSM but with unknown scale. The composite point cloud data generated by historical stereo images is fused with 2D map to normalize the scale. The same resolution of imagery with 15 cm/pixel GSD is used. The density of the generated point cloud is 40 points/m^2. The point cloud is back-projected to 2D imagery as DSM, with scale GSD of 15 cm/pixel. The DSM from stereo images is evaluated by using the testing dataset, the outcome is reported in Table 5.

Table 5. Monocular DSM estimation outperforms the stereo imagery-based method. Lower values of MAE and RMSE are better.

DSM Method	GSD	MAE (m) ↓	RMSE (m) ↓
OSI stereo	15 cm/pixel	2.36	4.90
IM2ELEVATION (with MI, patch adjustment)	15 cm/pixel	1.46	3.05

6. Discussion

We present a panel of detailed examples of DSM reconstruction obtained with our pipeline and from stereo imagery in Figure 11. The heat maps highlight the difference between our estimated

DSM and the corresponding ground truth DSM. We observe that our estimates present with less salt-and-pepper-type noise and succeed in extracting the outlines of the buildings on a par with the stereoscopic imagery input.

To investigate the potential of height reconstruction for visualization purposes, we now report the results of mesh reconstruction in Figure 12. Specifically, the estimated DSM is back-projected to a 3D point cloud using pixel georeferencing information and elevation value, and reconstructed to a 3D mesh by 2.5D Delaunay triangulation [57]. The points are triangulated in 2D space by using their row and column geolocation coordinates. The edges between vertices are inherited from pixels. Finally, the input aerial images are overlaid with the resulting building mesh. The Eye-Dome Lighting shader [58] is applied to facilitate depth visualization.

In Figure 13, we attempt to perform DSM estimation on a different image dataset. To this end, we collect Google Maps satellite images covering the same geographic extent as the crops in Figure 11 using Google API. We then perform inference on these without any additional retraining or fine-tuning of our CNN pipeline. We observe that the estimation results are substantially worse, which is due to different properties of the Google images as the result of a completely different (and not explicitly disclosed) preprocessing applied to these images. This highlights the necessity for pipeline fine-tuning when the image source changes. As demonstrated by the results reported in Section 5.2 in Potsdam and DFC2018 experiments, once retrained, the IM2ELEVATION pipeline is capable of producing state-of-the-art results.

In Figure 14 we used ArcGIS functionalities to postprocess the predicted DSM applying the footprint constraints (we use OSI-provided footprints data) via Local Government 3D Basemaps package (https://solutions.arcgis.com/local-government/help/local-government-scenes/get-started/). The simple roof shape can be reconstructed to CityGML [59] style from 2.5D and reach the level of detail 2, i.e., LoD2. It is worth of mentioning that the work from Alidoost et al. [35] reached the similar level of precision but they used additional data while training.

The training data used in the study covers solely urban areas of high density and lacks any examples of lower density or suburban scenes. Poor performance of our trained pipeline is therefore observed where buildings are sparsely distributed in the scene, see Figure 15. This performance drop is due to the absence/weakness of support information for height inference such as neighboring buildings and shadows. Much stronger presence of tall vegetation also requires training of the pipeline on a more relevant suburban dataset.

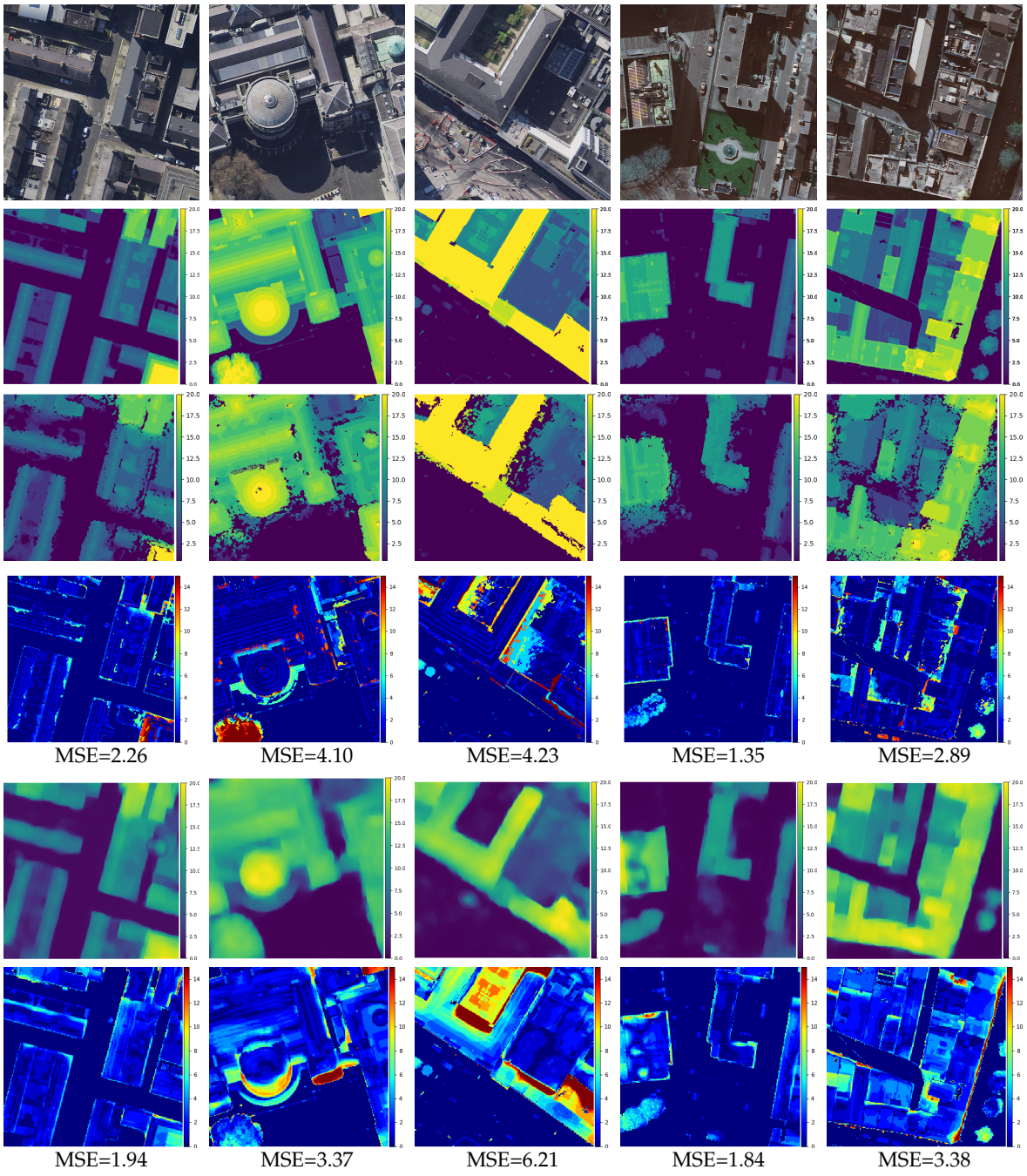


Figure 11. DSM from stereo imagery (source Ordnance Survey Ireland (OSI)) vs. DSM from IM2ELEVATION. From **top** to **bottom** (rows 1–6): (1) aerial image input, (2) ground truth DSM, (3) DSM generated from stereo images, (4) heat map of the difference between (2) and (3), (5) DSM from IM2ELEVATION, and (6) heat map of the difference between (2) and (5). Color scale is in meters. Note the strong impact of a tree canopy in the second column sample with IM2ELEVATION DSM reconstructing this element better.

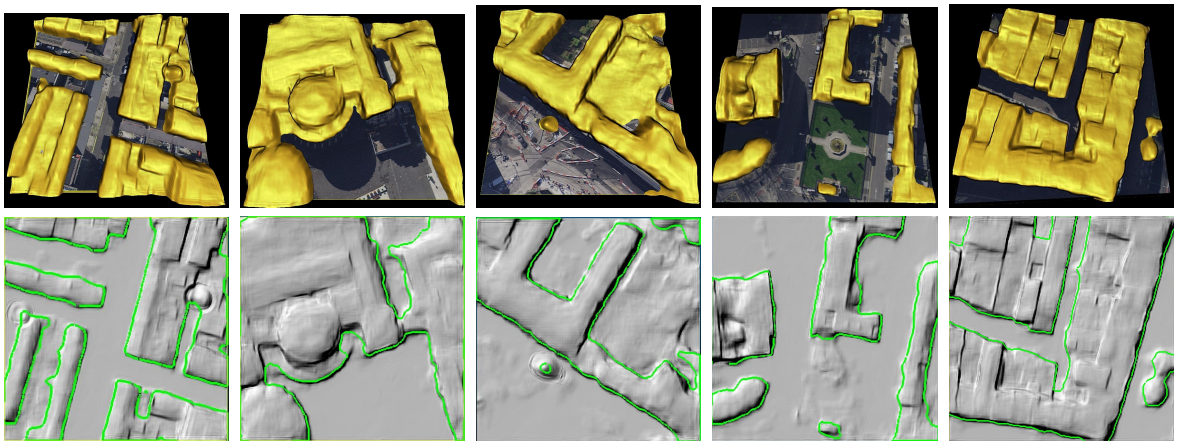


Figure 12. Further processing of our DSM estimates for mapping: 3D reconstruction and 2D building footprint detection. 3D mesh is obtained via Delaunay triangulation with shading.

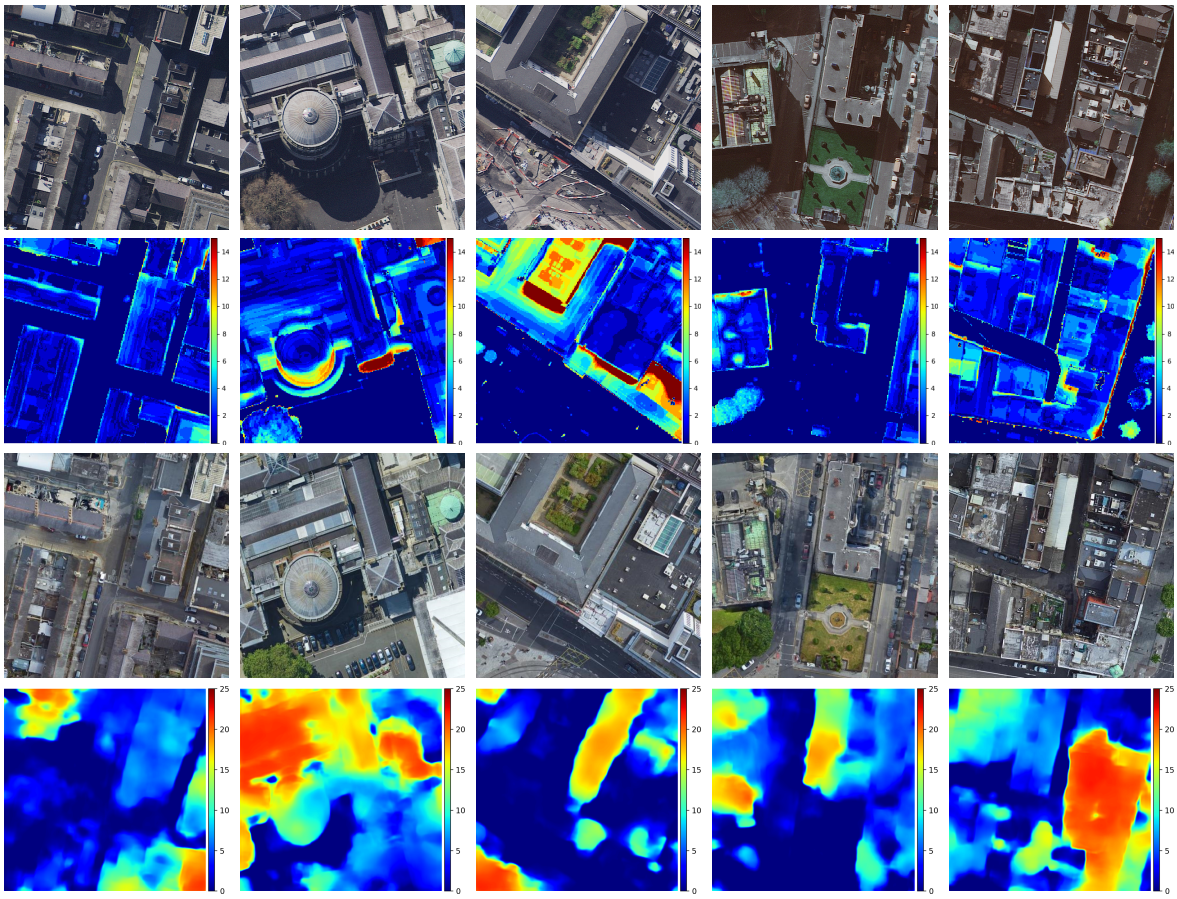


Figure 13. Comparison of DSM reconstruction (without retraining) from OSI aerial imagery (row 1) and Google Maps satellite images (row 3) and the corresponding error of estimations heat maps.

Figure 14. IM2ELEVATION inference results applied to roof profile estimation (LoD2).

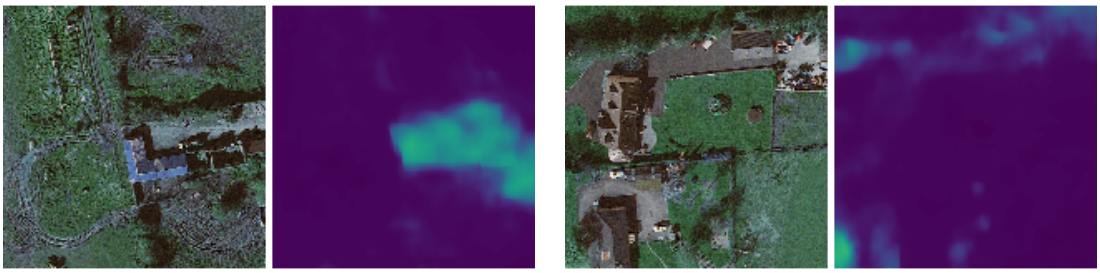


Figure 15. Examples of poor performance of DSM estimation on sparse scenes.

7. Conclusions

We have proposed a pipeline for inferring heights from aerial images using machine learning trained with two asynchronous heterogeneous datasets: aerial imagery and Lidar point cloud, where the latter is employed only during the training phase. We mitigated the impact of temporal and spatial mismatches in our custom dataset by removing areas that had changed between the acquisition dates (e.g., new building, etc.), and also proposed registration tools on the two data streams to correct small translation artifacts. Our pipeline manages to perform building height estimation on average within 1.5 m of the ground truth (which in this study is a Lidar point cloud with average height accuracy of 0.03 m), which outperforms the estimation results on stereo imagery. We demonstrated how the estimated DSM can be used for inferring 3D building shapes and 2D building footprints.

Future directions of research to enable higher level of accuracy in recovering heights and shapes of roofs include augmenting our training dataset with computer graphics simulated data (training set) and adding other input data streams such as street view imagery. IM2ELEVATION pipeline for DSM inference provides an alternative to stereo reconstruction when multiple aerial imagery is not available. Nevertheless, integration of the elements of this pipeline may provide highly relevant information in stereo reconstruction if such data is available. In the future we envisage to look into merging multiple DSMs (i.e., stereo-generated and AI-generated) to reduce noise and improve accuracy in addition to considering shape priors inherent in man-built infrastructure (e.g., planar surface).

Author Contributions: C.-J.L. developed the software and run the experiments; P.K. and G.K. prepared and provided the OSI data used in the experiments; C.-J.L., V.A.K., and R.D. conceived the experiments and wrote the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This research was conducted with the financial support of Ordnance Survey Ireland and the Science Foundation Ireland under Grant Agreement No. 13/RC/2106 at the ADAPT SFI Research Centre at Trinity College Dublin and Dublin City University. The ADAPT SFI Centre for Digital Media Technology is funded by Science Foundation Ireland through the SFI Research Centres Programme and is co-funded under the European Regional Development Fund (ERDF) through Grant 13/RC/2106.

Acknowledgments: We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPUs. The Dublin aerial optical imagery was provided by Ordnance Survey Ireland. DFC2018 dataset is from 2018 IEEE GRSS Data Fusion Contest. The Potsdam/Vaihingen dataset was provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF).

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

References

1. Bosch, M.; Foster, K.; Christie, G.; Wang, S.; Hager, G.D.; Brown, M. Semantic stereo for incidental satellite images. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019; pp. 1524–1532.
2. Schönberger, J.L.; Frahm, J.M. Structure-from-Motion Revisited. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016.
3. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
4. Hu, J.; Ozay, M.; Zhang, Y.; Okatani, T. Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019; pp. 1043–1051.
5. Laefer, D.F.; Saleh Abuwarda, A.V.V.; Truong-Hong, L.; Gharibi, H. 2015 Aerial Laser and Photogrammetry Survey of Dublin City Collection Record. 2015. Available online: https://geo.nyu.edu/catalog/nyu_2451_38684 (accessed on 21 August 2020). [CrossRef]
6. LIDAR Point Cloud UK. Available online: https://data.gov.uk/dataset/977a4ca4-1759-4f26-baa7-b566bd7ca7bf/lidar-point-cloud (accessed on 21 August 2020).
7. Malof, J.M.; Bradbury, K.; Collins, L.M.; Newell, R.G. Automatic detection of solar photovoltaic arrays in high resolution aerial imagery. *Appl. Energy* **2016**, *183*, 229–240. [CrossRef]
8. HERE Geodata Models offer global precise 3D dataset for 5G deployment. *Geo Week News*, 2 April 2020. Available online: https://www.geo-week.com/here-geodata-models-offer-global-precise-3d-dataset-for-deploying-5g/ (accessed on 21 August 2020).
9. Ahmad, K.; Pogorelov, K.; Riegler, M.; Ostroukhova, O.; Halvorsen, P.; Conci, N.; Dahyot, R. Automatic detection of passable roads after floods in remote sensed and social media data. *Signal Process. Image Commun.* **2019**, *74*, 110–118. [CrossRef]
10. Bulbul, A.; Dahyot, R. Social media based 3D visual popularity. *Comput. Graph.* **2017**, *63*, 28–36. [CrossRef]
11. Micusik, B.; Kosecka, J. Piecewise planar city 3D modeling from street view panoramic sequences. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 2906–2912.
12. Krylov, V.A.; Kenny, E.; Dahyot, R. Automatic Discovery and Geotagging of Objects from Street View Imagery. *Remote Sens.* **2018**, *10*, 661. [CrossRef]

13. Laumer, D.; Lang, N.; van Doorn, N.; Aodha, O.M.; Perona, P.; Wegner, J.D. Geocoding of trees from street addresses and street-level images. *ISPRS J. Photogramm. Remote Sens.* **2020**, *162*, 125–136. [CrossRef]
14. Liu, C.J.; Krylov, V.; Dahyot, R. 3D point cloud segmentation using GIS. In Proceedings of the 20th Irish Machine Vision and Image Processing Conference, Belfast, UK, 29–31 August 2018; pp. 41–48.
15. Byrne, J.; Connelly, J.; Su, J.; Krylov, V.; Bourke, M.; Moloney, D.; Dahyot, R. Trinity College Dublin Drone Survey Dataset. (Imagery, Mesh and Report), Trinity College Dublin. 2017. Available online: http://hdl.handle.net/2262/81836 (accessed on 21 August 2020).
16. Benedek, C.; Descombes, X.; Zerubia, J. Building Development Monitoring in Multitemporal Remotely Sensed Image Pairs with Stochastic Birth-Death Dynamics. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 33–50. [CrossRef] [PubMed]
17. Lafarge, F.; Descombes, X.; Zerubia, J.; Pierrot-Deseilligny, M. Structural Approach for Building Reconstruction from a Single DSM. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 135–147. [CrossRef] [PubMed]
18. Maggiori, E.; Tarabalka, Y.; Charpiat, G.; Alliez, P. Can Semantic Labeling Methods Generalize to Any City? The Inria Aerial Image Labeling Benchmark. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017.
19. Palmer, D.; Koumpli, E.; Cole, I.; Gottschalg, R.; Betts, T.A. GIS-Based Method for Identification of Wide Area Rooftop Suitability for Minimum Size PV Systems Using LiDAR Data and Photogrammetry. *Energies* **2018**, *11*, 3506. [CrossRef]
20. Song, X.; Huang, Y.; Zhao, C.; Liu, Y.; Lu, Y.; Chang, Y.; Yang, J. An Approach for Estimating Solar Photovoltaic Potential Based on Rooftop Retrieval from Remote Sensing Images. *Energies* **2018**, *11*, 3172. [CrossRef]
21. Saxena, A.; Sun, M.; Ng, A.Y. Make3D: Depth Perception from a Single Still Image. *AAAI* **2008**, *3*, 1571–1576.
22. Saxena, A.; Chung, S.H.; Ng, A.Y. Learning depth from single monocular images. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 4–7 December 2006; pp. 1161–1168.
23. Zhuo, W.; Salzmann, M.; He, X.; Liu, M. Indoor scene structure analysis for single image depth estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 8–10 June 2015; pp. 614–622.
24. Eigen, D.; Fergus, R. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 13–16 December 2015; pp. 2650–2658.
25. Liu, F.; Shen, C.; Lin, G. Deep convolutional neural fields for depth estimation from a single image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 8–10 June 2015; pp. 5162–5170.
26. Xu, D.; Ricci, E.; Ouyang, W.; Wang, X.; Sebe, N. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5354–5362.
27. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 8–10 June 2015; pp. 3431–3440.
28. Laina, I.; Rupprecht, C.; Belagiannis, V.; Tombari, F.; Navab, N. Deeper depth prediction with fully convolutional residual networks. In Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), Stanford, CA, USA, 25–28 October 2016; pp. 239–248.
29. Mal, F.; Karaman, S. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 1–8.
30. Eigen, D.; Puhrsch, C.; Fergus, R. Depth map prediction from a single image using a multi-scale deep network. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 2366–2374.
31. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, CA, USA, 3–8 December 2012; pp. 1097–1105.

32. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
33. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
34. Drozdzal, M.; Vorontsov, E.; Chartrand, G.; Kadoury, S.; Pal, C. The importance of skip connections in biomedical image segmentation. In *Deep Learning and Data Labeling for Medical Applications*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 179–187.
35. Alidoost, F.; Arefi, H.; Tombari, F. 2D Image-To-3D Model: Knowledge-Based 3D Building Reconstruction (3DBR) Using Single Aerial Images and Convolutional Neural Networks (CNNs). *Remote Sens.* **2019**, *11*, 2219. [CrossRef]
36. Mou, L.; Zhu, X.X. IM2HEIGHT: Height estimation from single monocular imagery via fully residual convolutional-deconvolutional network. *arXiv* **2018**, arXiv:1802.10249.
37. Amirkolaee, H.A.; Arefi, H. Height estimation from single aerial images using a deep convolutional encoder-decoder network. *ISPRS J. Photogramm. Remote Sens.* **2019**, *149*, 50–66. [CrossRef]
38. Srivastava, S.; Volpi, M.; Tuia, D. Joint height estimation and semantic labeling of monocular aerial images with CNNs. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 5173–5176.
39. Carvalho, M.; Le Saux, B.; Trouvé-Peloux, P.; Champagnat, F.; Almansa, A. Multitask Learning of Height and Semantics From Aerial Images. *IEEE Geosci. Remote Sens. Lett.* **2019**, *17*, 1391–1395. [CrossRef]
40. Ghamisi, P.; Yokoya, N. Img2dsm: Height simulation from single imagery using conditional generative adversarial net. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 794–798. [CrossRef]
41. Bittner, K.; d'Angelo, P.; Körner, M.; Reinartz, P. Dsm-to-lod2: Spaceborne stereo digital surface model refinement. *Remote Sens.* **2018**, *10*, 1926. [CrossRef]
42. Habib, A.; Ghanma, M.; Morgan, M.; Al-Ruzouq, R. Photogrammetric and LiDAR data registration using linear features. *Photogramm. Eng. Remote Sens.* **2005**, *71*, 699–707. [CrossRef]
43. Kwak, T.S.; Kim, Y.I.; Yu, K.Y.; Lee, B.K. Registration of aerial imagery and aerial LiDAR data using centroids of plane roof surfaces as control information. *KSCE J. Civ. Eng.* **2006**, *10*, 365–370. [CrossRef]
44. Peng, S.; Ma, H.; Zhang, L. Automatic Registration of Optical Images with Airborne LiDAR Point Cloud in Urban Scenes Based on Line-Point Similarity Invariant and Extended Collinearity Equations. *Sensors* **2019**, *19*, 1086. [CrossRef]
45. Zhang, W.; Zhao, J.; Chen, M.; Chen, Y.; Yan, K.; Li, L.; Qi, J.; Wang, X.; Luo, J.; Chu, Q. Registration of optical imagery and LiDAR data using an inherent geometrical constraint. *Opt. Express* **2015**, *23*, 7694–7702. [CrossRef]
46. Chen, H.; Xie, W.; Vedaldi, A.; Zisserman, A. AutoCorrect: Deep Inductive Alignment of Noisy Geometric Annotations. *arXiv* **2019**, arXiv:1908.05263.
47. Viola, P.; Wells, W.M., III. Alignment by maximization of mutual information. *Int. J. Comput. Vis.* **1997**, *24*, 137–154. [CrossRef]
48. Mastin, A.; Kepner, J.; Fisher, J. Automatic registration of LIDAR and optical images of urban scenes. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern recognition, Miami, FL, USA, 20–25 June 2009; pp. 2639–2646.
49. Parmehr, E.G.; Fraser, C.S.; Zhang, C.; Leach, J. Automatic registration of optical imagery with 3D LiDAR data using statistical similarity. *ISPRS J. Photogramm. Remote Sens.* **2014**, *88*, 28–40. [CrossRef]
50. Shannon, C.E. A mathematical theory of communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. [CrossRef]
51. Styner, M.; Brechbuhler, C.; Szckely, G.; Gerig, G. Parametric estimate of intensity inhomogeneities applied to MRI. *IEEE Trans. Med. Imaging* **2000**, *19*, 153–165. [CrossRef]
52. 2018 IEEE GRSS Data Fusion Contest. Available online: http://www.grss-ieee.org/community/technical-committees/data-fusion (accessed on 21 August 2020).
53. Xu, Y.; Du, B.; Zhang, L.; Cerra, D.; Pato, M.; Carmona, E.; Prasad, S.; Yokoya, N.; Hänsch, R.; Le Saux, B. Advanced multi-sensor optical remote sensing for urban land use and land cover classification: Outcome of the 2018 IEEE GRSS Data Fusion Contest. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2019**, *12*, 1709–1724. [CrossRef]

54. ISPRS Potsdam 2D Semantic Labeling Contest. Available online: http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-potsdam.html (accessed on 21 August 2020).
55. ISPRS Vaihingen 2D Semantic Labeling Dataset. Available online: http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-vaihingen.html (accessed on 21 August 2020).
56. Facciolo, G.; De Franchis, C.; Meinhardt-Llopis, E. Automatic 3D reconstruction from multi-date satellite images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 57–66.
57. Fortune, S. A sweepline algorithm for Voronoi diagrams. *Algorithmica* **1987**, *2*, 153. [CrossRef]
58. Boucheny, C. Visualisation Scientifique de Grands Volumes de Données: Pour une Approche Perceptive. Ph.D. Thesis, Joseph Fourier University, Grenoble, France, 2009.
59. Gröger, G.; Plümer, L. CityGML–Interoperable semantic 3D city models. *ISPRS J. Photogramm. Remote Sens.* **2012**, *71*, 12–33. [CrossRef]