

BUILDING OUTLINE EXTRACTION FROM AERIAL IMAGERY AND DIGITAL SURFACE MODEL WITH A FRAME FIELD LEARNING FRAMEWORK

Xiaoyu Sun¹, Wufan Zhao¹, Raian V. Maretto¹, Claudio Persello^{1,*}

¹ Dept. of Earth Observation Science, ITC, University of Twente, Enschede, The Netherlands – x.sun@student.utwente.nl, wufan.zhao@utwente.nl, r.v.maretto@utwente.nl, c.persello@utwente.nl

Commission II, WG II/6

KEY WORDS: Building Outline Delineation, Convolutional Neural Networks, Regularized Polygonization, Frame Field

ABSTRACT:

Deep learning-based semantic segmentation models for building delineation face the challenge of producing precise and regular building outlines. Recently, a building delineation method based on frame field learning was proposed by Girard et al., (2020) to extract regular building footprints as vector polygons directly from aerial RGB images. A fully convolution network (FCN) is trained to learn simultaneously the building mask, contours, and frame field followed by a polygonization method. With the direction information of the building contours stored in the frame field, the polygonization algorithm produces regular outlines accurately detecting edges and corners. This paper investigated the contribution of elevation data from the normalized digital surface model (nDSM) to extract accurate and regular building polygons. The 3D information provided by the nDSM overcomes the aerial images' limitations and contributes to distinguishing the buildings from the background more accurately. Experiments conducted in Enschede, the Netherlands, demonstrate that the nDSM improves building outlines' accuracy, resulting in better-aligned building polygons and prevents false positives. The investigated deep learning approach (fusing RGB + nDSM) results in a mean intersection over union (IOU) of 0.70 in the urban area. The baseline method (using RGB only) results in an IOU of 0.58 in the same area. A qualitative analysis of the results shows that the investigated model predicts more precise and regular polygons for large and complex structures.

1. INTRODUCTION

Building extraction has been active for decades due to the availability of a large amount of very high-resolution remote sensing data and the need for detailed information of small-scale objects in multiple applications. Precisely extracting the building boundaries is of utmost importance for producing cadastral and topographic maps and applications in urban planning and management. With the rise of deep learning, deep convolutional neural networks (CNNs) based models became the dominant approach in building extraction. CNNs have outperformed traditional methods based on spectral and geometric features. However, accurately extracting buildings is still challenging for several reasons: (i) Buildings have various sizes, geometrical complexity, and spectral responses across the bands. (ii) Trees or their shadows often obscure them. (iii) The high intra-class and low inter-class variation of building objects in high-resolution remotely sensed images make it hard to extract the buildings' spectral and geometrical features (Huang, Zhang, Xin, Sun, & Zhang, 2019).

Automatically delineating regularized building boundaries as polygons is a promising direction. Most deep learning techniques focus on producing the binary segmentation map by the neural network. However, applications based on geographic information systems (GIS) require a vector representation of polygon objects. Raster maps demand complicated and expensive post-processing to obtain polygons (Girard et al., 2020). PolyMapper, proposed by Li et al. (2019), is an end-to-

end deep learning architecture that can automatically delineate small buildings' boundaries. Zhao, Persello, & Stein (2021) upgrade the feature extractor and extraction module of Polymapper and improve its performance. To differentiate buildings from their complex background in Very High Resolution (VHR) remotely sensed images, a boundary refinement block (BRB) is introduced to amplify the distinction of features. However, such a method's performance decreases significantly with large buildings, resulting in less accurate outlines than Mask R-CNN (Li et al., 2019). Moreover, it cannot deal with polygons with holes. To better regularize the complex building, such as buildings with holes, Girard et al. (2020) trained an FCN to learn the interior map, edge map, and a frame field aligned with the building outline tangents. Then the frame field and interior map are used in their polygonization algorithm to produce regular and accurate building polygons.

Due to the limitations of optical sensors and the availability of multimodal data, the use of data fusion to improve building extraction accuracy has been an important research field for many years. LiDAR sensors have a different imaging mechanism that makes them able to penetrate the clouds and sparse vegetation. Hence, elevation models derived from LiDAR data can significantly alleviate the performance degradation in building delineation caused by the lack of height information in optical images (Hong et al., 2020). Digital Surface Model (DSM) and nDSM are popular options to provide 3D information in data fusion. Different combinations of fusion data and network architectures have been proposed to extract the building footprint.

* Corresponding author

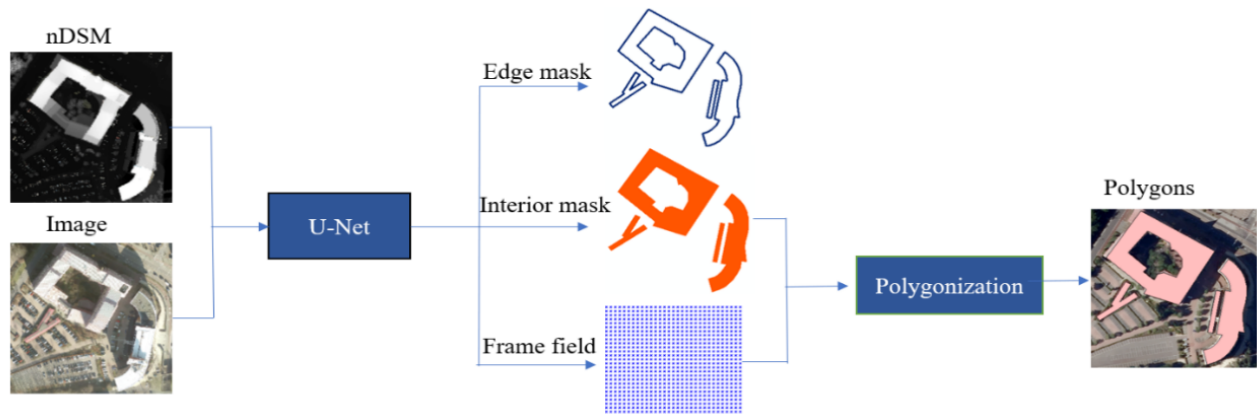


Figure 1. The workflow of the investigated frame-field method for building delineation fusing nDSM and RGB data. Adapted from Girard et al., (2020).

Bittner et al., (2018) proposed a Fused-FCN4s network that used three branches to take the RGB, nDSM, and panchromatic (PAN) channels as input for each branch separately. Hong et al., (2020) tested different fusion modules and found that compactness-based fusion networks (including encoder-decoder fusion strategy and a newly proposed cross fusion) present a performance superior to others in blending multimodal features.

State-of-the-art methods (Girard et al., 2020; Li et al., 2019) only take RGB imagery as input. The fusion of aerial images and nDSM could provide more information and help overcome problems in building extraction like low contrast with its neighbour and obscurity caused by shadows. We follow this research line by introducing nDSM and RGB data fusion to the framework to improve building outline accuracy. The three main contributions of this study are:

- 1) We introduce the nDSM into the model, use the fusion of VHR images and 3D information to optimize information extraction in building segmentation.
- 2) We experimentally investigate our approach on a new data set acquired in the city of Enschede, the Netherlands.
- 3) We evaluated the performance of the considered methods adopting different metrics assessed at pixel, object and polygon level.

2. METHODOLOGY

2.1 Overview

Girard et al. (2020) proposed a framework based on an FCN to perform multi-task learning for pixel-wise segmentation. In particular, a frame field aligned with the object tangents is learned at every pixel of the image and used by the polygonization algorithm to create regular polygons aligned to the reference data, especially for the complex buildings with a slanted wall. A frame field consists of two pairs of vectors, each pair with π symmetry (Vaxman et al., 2016). Figure 1 shows the investigated framework, including two main parts. The first part is the U-Net, which takes RGB images and nDSM data as input for multi-task learning. The second part is the polygonization algorithm, which takes the segmentation and frame field produced by FCN to generate the polygons. With the height information provided by the nDSM, the segmentation and frame field generated by the network are improved. In the polygonization algorithm, the segmentation is vectorized and simplified into polygons using the direction information from the

frame field. Therefore, the predicted footprint in the polygon is also improved.

2.2 Frame field learning

Unlike the original framework, we use ResNet-101 as the encoder instead of U-Net16. As the network layer becomes deeper, the performance usually starts to decrease. He et al. (2016) proposed the residual networks with the identity shortcuts to address the degradation problems. These shortcuts skip one or more layers to performs identical mapping and add their output to the stacked layers.

The first layer of the network is extended to support taking input images with four channels. Then the output features of the backbone are fed into two branches with a shallow structure. The specific structure is shown in Figure 2. The edge mask and interior mask are produced by one network as two channels of an image. The frame field is produced by another network as an image of four channels.

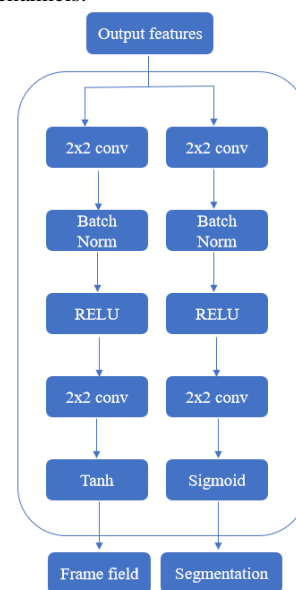


Figure 2. The two branches to produce segmentation and frame field.

The model was trained in a supervised way. The polygons in reference footprints are rasterized in the pre-processing part of the algorithm to generate reference edge mask and interior mask.

For the frame field, the reference is an angle calculated from the segments of the footprints. These related tasks help the model to focus on the important and representative feature extracted from the input data. The combined loss functions constrain these tasks to make them consistent with each other.

The interior map and frame field are then input to the polygonization method. First, an initial contour is extracted from the interior map by marching squares (Lorenzen & Cline, 1987). Then they are optimized by an active contours model (ACM)(Kass & Witkin, 1988) to make them more aligned to the frame field. Before simplification, corners are found with the direction information of the frame field. Then contours are split at corners into edges. Edges are simplified to reduce the number of vertices and produce a regular shape. In this phase, all the vertices within the tolerance distance, controlled by the tolerance parameter, are removed from the original edges.

2.3 Loss Function

The total loss function combines multiple loss functions for the different learning tasks: 1) segmentation, 2) frame field, and 3) coupling losses. H and W is the height and width of input image, respectively. The segmentation loss consists of a cross-entropy function for edge mask and interior mask, given by the equation below.

$$L_{BCE}(\hat{y}, y) = \frac{1}{HW} \sum_{x \in I} \hat{y}(x) \cdot \log(y(x)) + (1 - \hat{y}(x)) \cdot \log(1 - y(x)) \quad (1)$$

where L_{BCE} is the cross-entropy loss applied to the interior and the edge outputs of the model, respectively. The frame field is an essential element in the polygonization algorithm. The output frame field contains four channels, each two for the two complex coefficients $C_0, C_2 \in \mathbb{C}$. They define an equivalence class corresponding to a frame field. The reference is an angle $\theta_\tau \in [0, \pi)$, it is the tangent vector of the building contour.

$$L_{align} = \frac{1}{HW} \sum_{x \in I} \hat{y}_{edge}(x) f(e^{i\theta_\tau}; C_0(x), C_2(x))^2 \quad (2)$$

$$L_{align90} = \frac{1}{HW} \sum_{x \in I} \hat{y}_{edge}(x) f(e^{i\theta_\tau \perp}; C_0(x), C_2(x))^2 \quad (3)$$

$$L_{smooth} = \frac{1}{HW} \sum_{x \in I} (\|\nabla C_0(x)\|^2 + \|\nabla C_2(x)\|^2) \quad (4)$$

Where θ_τ is the direction of vector τ , and $\tau^\perp = \tau - \frac{\pi}{2}$. The L_{align} makes the frame field more aligned with the tangent of the line segment of polygons. $L_{align90}$ prevents the frame field from collapsing into a line field. L_{smooth} produces a smooth frame field. Because these outputs are closely related and represent different information of the building footprints, there are the following functions to make them compatible with each other.

$$L_{int\ align} = \frac{1}{HW} \sum_{x \in I} f(\nabla y_{int}(x); C_0(x), C_2(x))^2 \quad (5)$$

$$L_{edge\ align} = \frac{1}{HW} \sum_{x \in I} f(\nabla y_{edge}(x); C_0(x), C_2(x))^2 \quad (6)$$

$$L_{int\ edge} = \frac{1}{HW} \sum_{x \in I} \max(1 - y_{int}(x), \|\nabla y_{int}(x)\|_2 \cdot \|\nabla y_{int}(x)\|_2 - y_{edge}(x)) \quad (7)$$

Where $L_{int\ align}$ and $L_{edge\ align}$ constrain interior mask y_{int} and edge mask y_{edge} aligned with the frame field. The $L_{int\ edge}$ is to make the interior and edge mask compatible with each other.

3. EXPERIMENTAL RESULTS

3.1 Datasets and Evaluation Metrics

3.1.1 Datasets

We performed experiments in the municipality of Enschede, the Netherlands. The dataset contains three parts. 1) A VHR true-ortho aerial image with 0.25 m spatial resolution provided by Kadaster¹. The image of the study area is part of the nationwide summer flight. A Web Map Service (WMS) of this dataset is publicly available on PDOK², a portal website hosting open datasets from the government with current geo-information. 2) An nDSM obtained by subtracting the digital terrain model (DTM) from the DSM, then resampled to 0.25 m. The DTM and DSM with 0.5 m resolution are publicly available. The DTM has 'no-data' values in built-up areas filled using the QGIS 'fill nodata' tool with a maximum distance of 1000 pixels. AHN³ is the digital elevation map for all of the Netherlands. AHN3 dataset is acquired in the 3rd acquisition period (2014-2019), and the DTM and DSM are derived from point cloud based on the Squared IDW method with 0.5 m resolution. The mean point density of AHN3 is 8-10 points/m². The LiDAR point clouds and DSM are shown in Figure 3. 3) Building footprints are obtained using publicly available geodata combining small buildings from the BAG⁴ with larger ones from BRT⁵. The BAG is part of the government system of key registers. Municipalities are source holders of the BAG. The BRT is a collection of digital topographical data on different scales. Buildings from the TOP10NL product were used in this research, which is topographical data suitable for the scales 1:5000-1:25000. Example images and the corresponding label are shown in Figure 4. The nDSM is stacked as a 4th channel on top of the RGB images, producing a composite image.

We consider two datasets with different spatial extent. One comprises the entire study area. Another one only includes the urban area. The extent and distribution of tiles are shown in Figure 5. For each dataset, tiles are extracted from the aerial image (RGB) and composite image (RGB + nDSM) with the same location and size. The dataset details are shown in Table 1.

Dataset	Type	Number of tiles	Number of buildings	Ratio
Entire study area	training	1107	26383	0.6
	validation	369	8184	0.2
	test	369	8392	0.2
Urban area	training	579	27268	0.7
	validation	82	3924	0.1
	test	165	7803	0.2

Table 1. Information of training, validation, and test set for the two study areas. The size of each tile is 1024×1024 pixels.

¹ Kadaster (The Netherlands' Cadastre, Land Registry and Mapping Agency)

² PDOK (the Public Services On the Map), <https://www.pdok.nl/>

³ AHN ((Het Actueel Hoogtebestand Nederland)

⁴ BAG (Basisregistratie Adressen en Gebouwen)

⁵ BRT (Basisregistratie Topografie)

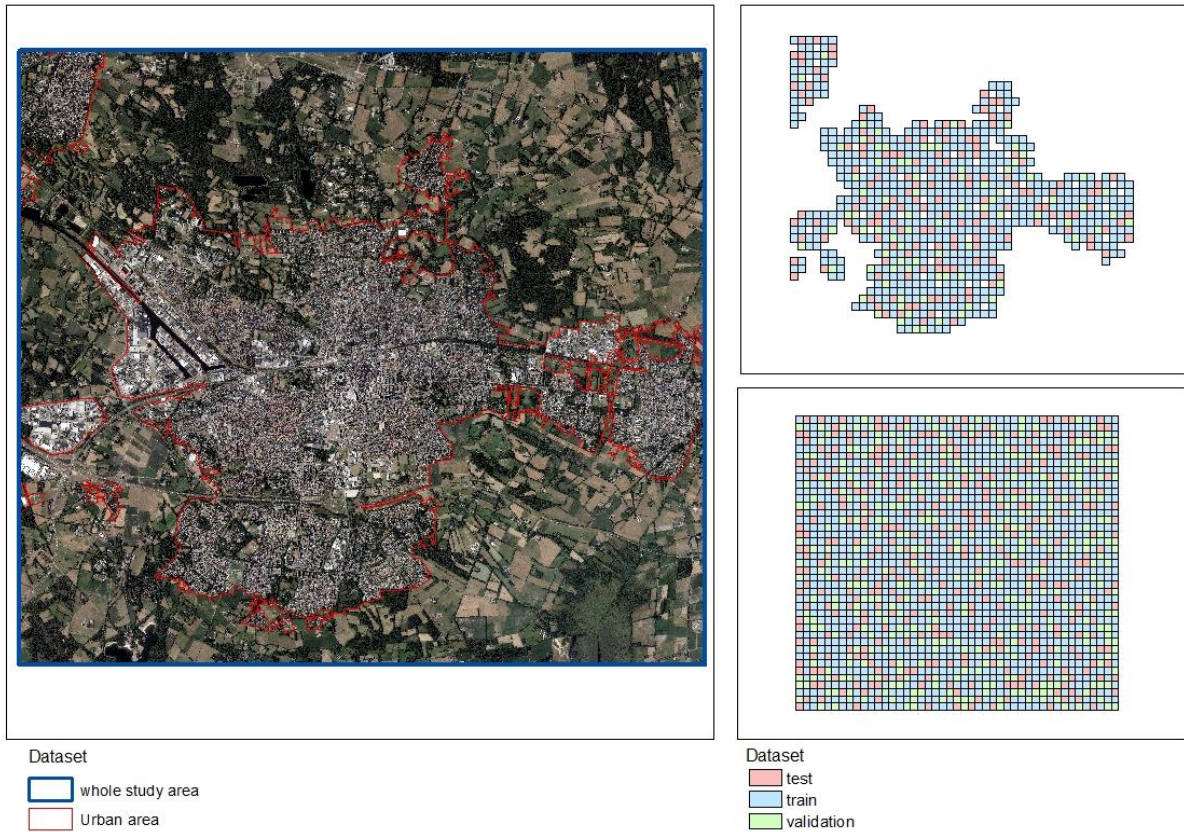


Figure 5. The entire study area is the whole image of the city of Enschede; the urban area is denoted by the red polygons (right). The right side shows the tile distribution for the urban area (upper right), and the entire study area (lower right).

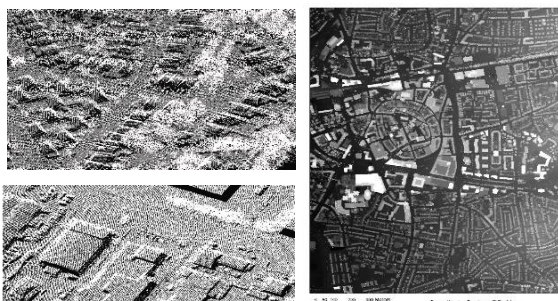


Figure 3. Sample data of LiDAR point clouds (left) and the derived DSM with 0.5 meters of spatial resolution (right)



Figure 4. Sample data of the building footprints (left) and an aerial image of the represented area (right).

3.1.2 Evaluation Metrics

Pixel-level metrics. For evaluating the results, we used the mean Intersection over Union (IoU). IoU is computed by dividing the intersection area by union area of a predicted segmentation (p) and a ground-truth (g) at the pixel level.

$$IoU = \frac{area(p \cap g)}{area(p \cup g)} \quad (8)$$

Object-level metrics. Building delineation is closely related to object segmentation, so we introduced mean Average Precision (AP) and mean Average Recall (AR) in Common Objects in Context (COCO) measures to evaluate the result. They help determine whether a building was extracted correctly, and whether a predicted building actually exists. AP and AR are calculated based on multiple Intersection over Union (IoU). There are 10 IoU thresholds ranging from 0.50 to 0.95 with 0.05 steps. AP and AR are the average value of all precisions and recalls calculated over 10 IoU categories. The metrics are usually applied on segmentation mask in COCO format. We followed the same standards of the metric but applied to building polygons directly. To be specific, the IOU calculation is based on polygons.

Polygon-level metrics. Besides the COCO metrics, Polygons and line segments measurement (PoLiS) were introduced to evaluate the similarity of the predicted polygons with corresponding reference polygons. It accounts for positional and shape differences by considering polygons as a sequence of connected edges instead of only point sets (Avbelj, Muller, & Bamler, 2014). We used this metric to evaluate the quality of the predicted

polygon. We first filter the polygons with $\text{IoU} \geq 0.5$ to find the prediction polygons and the corresponding reference polygons. The metric express as follows:

$$p(A, B) = \frac{1}{2q} \sum_{a_j \in A} \min_{b \in \partial B} \|a_j - b\| + \frac{1}{2r} \sum_{b_k \in B} \min_{a \in \partial A} \|b_k - a\| \quad (9)$$

where $p(A, B)$ is defined as the average of the distances between each vertex $a_j \in A, j = 1, \dots, q$, of A and its closest point $b \in \partial B$ on polygon B , plus the average of distances between each vertex $b_k \in B, k = 1, \dots, r$, of B and its closest point $a \in \partial A$ on polygon A . The closest point is not necessarily a vertex, it can be a point on edge. $(1/2q)$ and $(1/2r)$ are normalization factors to quantify the overall average dissimilarity per point.

3.2 Implementation Details

The model was trained with the following settings: Adam optimizer with a batch size $b = 4$ and an initial learning rate of 0.001. It applies an exponential decay to the learning rate with a decay rate of 0.99, the max epoch is set to 200. The network is implemented using PyTorch 1.4. The training and testing are performed on a single NVIDIA Tesla P100 GPU. We set several values (0.125, 1, 5, 7) for tolerance parameter in the polygonization method.

3.3 Results and Discussion

We compared results obtained on the test set of aerial images (RGB) and composite images (RGB + nDSM) for the entire study area and for the urban area, respectively. There are two results for each study area: 1) aerial images with RGB channels, and 2) multi-band images with nDSM as the additional channel. To ensure a fair comparison of the two models, the configurations are kept the same except for the input data.

3.3.1 Quantitative analysis

Table 2 shows the quantitative results obtained using the composite images (RGB + nDSM) and the single aerial images (RGB). For the entire study area, both mean IoUs are about 50%, with the experiment over the composite image performing slightly better. The main part of the framework is training the FCN to learn the segmentation and frame field of the buildings. Since the entire study area contains both urban areas and rural areas, even though the polygons have the similar ratio as the tiles in training, test, validation set, there are still some tiles with very few buildings or even without buildings. Girard et al., (2020) used the Inria dataset, which covers a larger extent and has all tiles extracted from urban settlements such as cities and towns (Maggiori, Tarabalka, Charpiat, & Alliez, 2017). Based on the difference between our dataset and the Inria dataset, we may hypothesize that the model needs more polygons in the training set to better learn the buildings' characteristics outside the city centres.

Table 3 shows that, for the urban area, The mean IoU achieved on the composite image test set was 70%, against 58% achieved for the test set of RGB image. This shows that the addition of the nDSM led to an improvement of 12% on the mean IoU, demonstrating that the model benefited from the data fusion. The higher average precision obtained using composite images shows that height information could help to reduce false positives, and

higher average recall shows it helps prevent missing the real buildings on the ground.

In terms of the similarity of the polygons, the PoLiS distance achieved on the composite image test set was 0.87, considerably smaller than 1.22 for the RGB image. Because the smaller PoLiS distance means the smaller dissimilarity, showing that the nDSM improves the similarity of the predicted polygons with the reference data.

Table 1 shows fewer tiles and more polygons in the urban area dataset than the entire city. The mean IoU achieved in the urban area was higher than that achieved in the entire city. It was 10% higher for the RGB images and 19% higher for the composite image. Hence we may deduce that the model performs better with the dense urban area.

Dataset	Data type	Mean IOU	mAP	mAR	PoLiS
Entire City	RGB images	0.48	0.04	0.09	1.06
	RGB + nDSM	0.51	0.05	0.12	1.33

Table 2. Extraction results on the entire study area of the Enschede dataset. The mAP, mAR and PoLiS are calculated on the polygons with 1 pixel tolerance for polygonization.

Dataset	Data type	Mean IOU	mAP	mAR	PoLiS
Urban area	RGB images	0.58	0.06	0.20	1.22
	RGB + nDSM	0.70	0.20	0.37	0.87

Table 3. Extraction results on the urban area of the Enschede dataset. The mAP, mAR and PoLiS are calculated on the polygons with 1 pixel tolerance for polygonization.

3.3.2 Qualitative analysis

Figure 6 compares the polygons predicted by the two models and the corresponding reference. The polygons obtained using the composite images are more aligned with the reference data and with fewer false positives than those obtained from RGB images only. The performance gain is particularly visible for big buildings with complex structures and the building with holes. Less false positives are observed for small buildings. In addition, the polygons of big buildings are more regular than the small ones in dense urban areas. In summary, the nDSM improved building outlines' accuracy, resulting in better-aligned building polygons and preventing false positives.

Figure 7 compares the predicted polygon on aerial image (RGB) with that on composite image (RGB+nDSM), showing that only with spectral information, the model cannot differentiate nearby buildings. This results in the predicted polygon on aerial image (RGB) corresponding to several individual buildings. In addition, the part of the road on the left side of the building is considered to be a building.

Figure 8 compares the predicted polygon with different tolerance levels. For the sample building, the increase of tolerance results in the decrease of the number of vertices. Table 4 shows that with the increase of tolerance, the PoLiS increase too, which means dissimilarity of the predicted polygon and reference polygon increase. Compared to the polygon with 1 pixel tolerance, some changes also happen to the shape of the polygon with 7 pixel

tolerance. In the lower right part of the polygon, as the vertices are simplified, the edge deviates from the ground truth too.

Polygon	b	c	d	e	f
tolerance		0.125	1	5	7
PoLiS		0.866	0.873	0.912	1.021
vertices	23	542	100	54	51

Table 4. Example polygon with different tolerance and number of vertices. The columns b, c, d, e, f correspond to the polygons (b), (c), (d), (e), (f) in Figure 8.

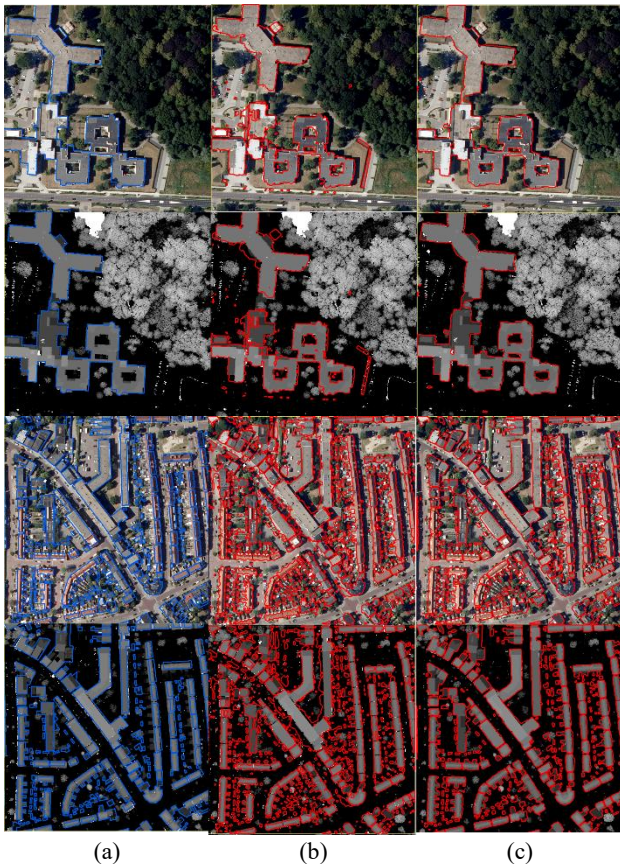


Figure 6. Results on the urban area test dataset. The background is the aerial image and the corresponding nDSM. The predicted polygons produced with 1 for the tolerance parameter of the polygonization method. From left to right: (a) Reference building footprints, (b) Predicted polygons on aerial images (RGB), (c) Predicted polygons on composite images (RGB + nDSM)

4. CONCLUSIONS

In this study, we investigated an automatic strategy for building outline polygon extraction fusing VHR images and a nDSM model, respectively. By adding the 3D information from the nDSM, the model based on the frame field learning was improved in terms of accuracy and regularity. The comparison against the results achieved only with the aerial image demonstrates that fusing those data helps in differentiating buildings from its surrounding, which results in polygons being more aligned with the reference boundaries. Our further study will focus on the following parts: 1) explore different fusion

strategies, 2) refining the training strategy, 3) explore and compare against other polygonization methods.

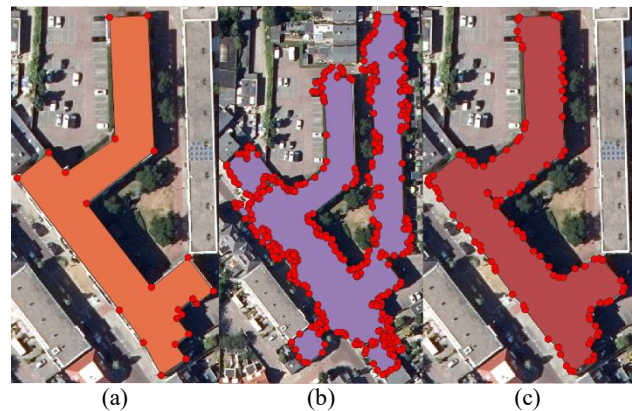


Figure 7. Results on the urban area test dataset. The predicted polygons produced with 1 pixel for the tolerance parameter of the polygonization method. From left to right: (a) Reference building footprints, (b) Predicted polygon on aerial images (RGB), (c) Predicted polygon on composite images (RGB + nDSM)

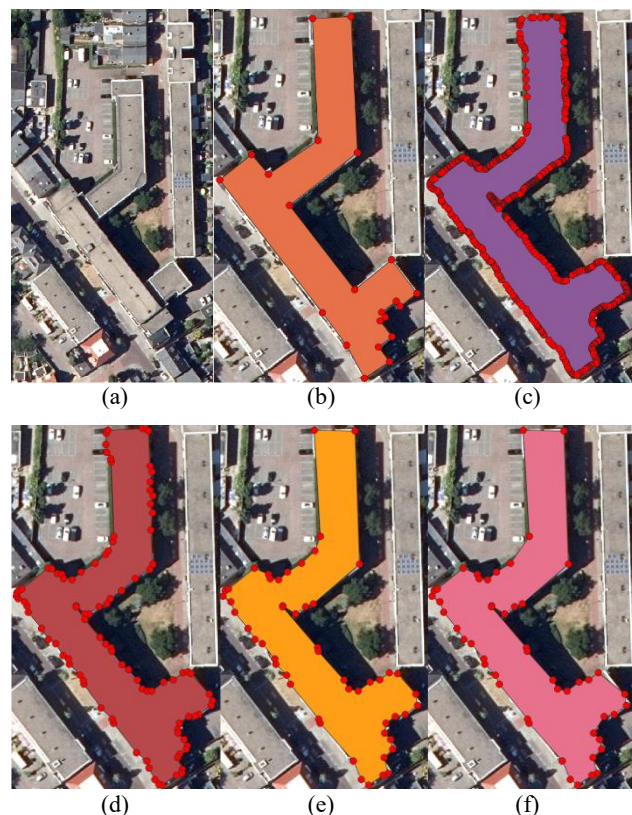


Figure 8. Example polygon obtained with different tolerance values using the composite images (RGB + nDSM) for urban area dataset: (a) Aerial image, (b) Reference building footprints, (c) Predicted polygon with tolerance 0.125 pixel, (d) 1 pixel, (e) 5 pixel, (f) and 7 pixel.

ACKNOWLEDGEMENTS

Sincere thanks to Kadaster for supporting this academic research with helpful support in data preparation and pre-processing.

REFERENCES

- Avbelj, J., Müller, R., & Bamler, R. (2014). A metric for polygon comparison and building extraction evaluation. *IEEE Geoscience and Remote Sensing Letters*, 12(1), 170-174.
- Bittner, K., Adam, F., Cui, S., Körner, M., & Reinartz, P. (2018). Building footprint extraction from VHR remote sensing images combined with normalized DSMs using fused fully convolutional networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(8), 2615-2629.
- Girard, N., Smirnov, D., Solomon, J., & Tarabalka, Y. (2020). Polygonal Building Segmentation by Frame Field Learning. *arXiv preprint arXiv:2004.14875*.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).
- Hong, D., Gao, L., Yokoya, N., Yao, J., Chanussot, J., Du, Q., & Zhang, B. (2020). More diverse means better: Multimodal deep learning meets remote-sensing imagery classification. *IEEE Transactions on Geoscience and Remote Sensing*.
- Huang, J., Zhang, X., Xin, Q., Sun, Y., & Zhang, P. (2019). Automatic building extraction from high-resolution aerial images and LiDAR data using gated residual refinement network. *ISPRS journal of photogrammetry and remote sensing*, 151, 91-105.
- Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. *International journal of computer vision*, 1(4), 321-331.
- Li, Z., Wegner, J.D. and Lucchi, A., 2019. Topological map extraction from overhead images. In *Proceedings of the IEEE International Conference on Computer Vision* pp. 1715-1724.
- Lorensen, W. E., & Cline, H. E. (1987). Marching cubes: A high resolution 3D surface construction algorithm. *ACM siggraph computer graphics*, 21(4), 163-169.
- Maggiori, E., Tarabalka, Y., Charpiat, G., & Alliez, P. (2017, July). Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (pp. 3226-3229). IEEE.
- Vaxman, A., Campen, M., Diamanti, O., Panozzo, D., Bommers, D., Hildebrandt, K., & Ben - Chen, M. (2016, May). Directional field synthesis, design, and processing. In *Computer Graphics Forum* (Vol. 35, No. 2, pp. 545-572).
- Zhao, W., Persello, C., & Stein, A. (2021). Building outline delineation: From aerial images to polygons with an improved end-to-end learning framework. *ISPRS Journal of Photogrammetry and Remote Sensing*, 175, 119-131.