Building segmentation from airborne vhr images using mask r-cnn

Zhou, K.; Chen, Y.; Smal, I.; Lindenbergh, R.

DOI
10.5194/isprs-archives-XLII-2-W13-155-2019

Publication date
2019

Document Version
Final published version

Published in
International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives

BUILDING SEGMENTATION FROM AIRBORNE VHR IMAGES USING MASK R-CNN

K. Zhou[1, *], Y. Chen[2], I. Smal[1], R. Lindenbergh[1]

[1] Dept. of Geoscience and Remote Sensing, Delft University of Technology, the Netherlands
{k.zhou-1, i.v.smal, r.c.lindenbergh}@tudelft.nl
[2] Dept. of Computational Science and Engineering, Delft University of Technology, the Netherlands
y.chen-35@student.tudelft.nl

Commission II, WG VI/4

KEY WORDS: 3D building model, VHR image, building segmentation, different scale of building, edge, Mask R-CNN, FPN, RPN, FCN

ABSTRACT:

Up-to-date 3D building models are important for many applications. Airborne very high resolution (VHR) images often acquired annually give an opportunity to create an up-to-date 3D model. Building segmentation is often the first and utmost step. Convolutional neural networks (CNNs) draw lots of attention in interpreting VHR images as they can learn very effective features for very complex scenes. This paper employs Mask R-CNN to address two problems in building segmentation: detecting different scales of building and segmenting buildings to have accurately segmented edges. Mask R-CNN starts from feature pyramid network (FPN) to create different scales of semantically rich features. FPN is integrated with region proposal network (RPN) to generate objects with various scales with the corresponding optimal scale of features. The features with high and low levels of information are further used for better object classification of small objects and for mask prediction of edges. The method is tested on ISPRS benchmark dataset by comparing results with the fully convolutional networks (FCN), which merge high and low level features by a skip-layer to create a single feature for semantic segmentation. The results show that Mask R-CNN outperforms FCN with around 15% in detecting objects, especially in detecting small objects. Moreover, Mask R-CNN has much better results in edge region than FCN. The results also show that choosing the range of anchor scales in Mask R-CNN is a critical factor in segmenting different scale of objects. This paper provides an insight into how a good anchor scale for different dataset should be chosen.

1. INTRODUCTION

Up-to-date 3D building models are crucial for many applications, such as water management, flooding simulation and urban planing. The rich geometric and spectral information in very high resolution (VHR) aerial images, which are often updated annually, gives an opportunity to create an up-to-date 3D model. However, due to the complexity of texture of building rooftops in VHR images, it is difficult to extract buildings. Convolutional neural networks (CNN) draw increasingly attention in interpreting VHR images (Yuan, 2018, Marmanis et al., 2018) due to their ability to automatically learn the most useful features for classification, instead of hand-crafting features manually. However, there are still several problems that need to be solved in order to utilize CNNs for building extraction. Firstly, neural networks (NN) are often data hungry and domain specific (Wang et al., 2017). The state-of-the-art performance of deep NNs is mostly due to training on large-scale benchmark datasets. However, benchmarks for VHR aerial images are limited and NNs trained on one dataset do not generalize well to similar types of image data. Secondly, instead of texture complexity, buildings often have diverse sizes, introducing also a scale problem. If small patches (for training) are selected, they tend to cover only parts of large buildings. The complete features of buildings can hardly be captured. If large patches are selected, the coarse resolution of the output from CNN due to pooling, which intends to extract high level features, is prone to losing small objects (Yuan, 2018, Ren et al., 2018). Thirdly, for similar reasons, low level features (e.g.,edges), often disappear in the higher level features as a large receptive field is used. Due to the deep layers of CNNs, if these features are not combined efficiently, edges are often poorly detected.

Topographic maps have been used to generate training samples automatically for VHR images (Maggiori et al., 2017, Kaiser et al., 2017, Chen et al., 2018). However, the topographic maps are often mismatched with images due to time differences. Transfer learning shows successful result in VHR images by fine tuning networks trained in ImageNet and COCO with less training samples.

Object-based image analysis using image pyramids has been proven by many researches to demonstrate good results in recognizing different scales of objects in VHR images using hand-engineered multi-scale features (Blaschke, 2010). The advantage is that image pyramids produce multi-scale features with strong semantic meanings in each level. For example, large objects, such as forest, are better to be classified in the coarse level of the image, while small objects, such as cars, are better to be classified in the finer level of the image. However, image pyramids are often only applied to CNNs for testing on ImageNet and COCO benchmark dataset (He et al., 2016). Training with image pyramids is often not feasible in term of memory of modern GPUs (Lin et al., 2017). In fact, the downsampling steps from pooling or strided convolution in deep layers of CNN already produce different level of features. The low level features are semantically week features, such as edges and boundaries, while the high level features are semantically strong features with contextual information. Fully convolutional networks (FCN) (Long et al., 2015) and U-net (Ronneberger et al., 2015) use the skip

[*] Corresponding author

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLII-2-W13-155-2019 | © Authors 2019. CC BY 4.0 License.

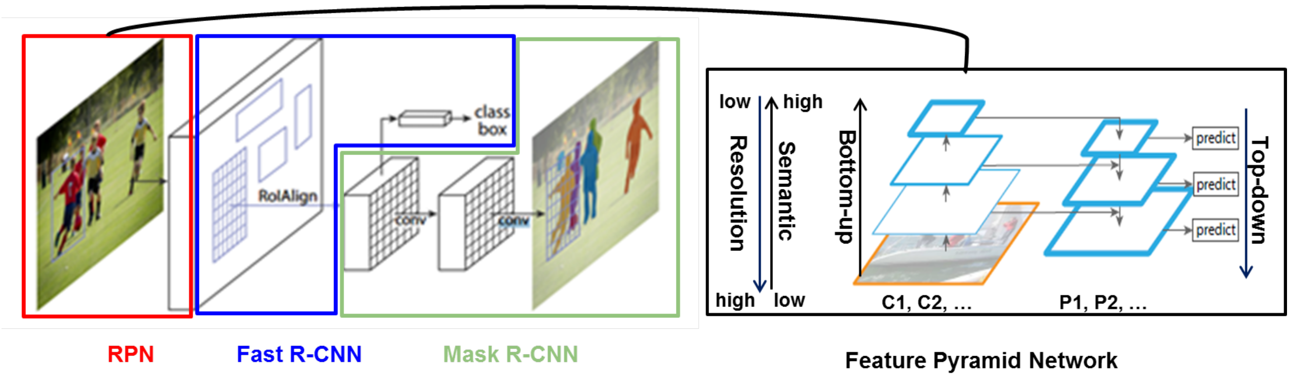

Figure 1. The framework of Mask R-CNN (He et al., 2017, Lin et al., 2017), which combines FPN, RPN and Fast R-CNN.

layer techniques to merge high and low resolution level feature maps for semantic segmentation. However, different scales of objects are predicted from a single high level features map of a fine resolution. (Yuan, 2018, Marmanis et al., 2018, Bittner et al., 2018) reported that the FCN with the most impact on semantic segmentation (Long et al., 2015) do not perform well in detecting building edges from VHR images.

Feature pyramid network (Lin et al., 2017) was proposed to create a feature pyramid with strong semantics in each level. It adopts a top-down architecture to merge high level feature maps with low level feature maps iteratively to create semantic rich feature maps at all levels for independent predictions. Mask R-CNN (He et al., 2017) utilizes region proposal network (Ren et al., 2015) to select the optimal level feature maps from pyramid for each region (object) detected. Further, Fast R-CNN (Girshick, 2015) is borrowed to refine regions and classify the region. In each region of interest (RoI), an FCN is applied to the optimal feature map to predict a pixel-wise segmentation/classification mask. The low level features of small objects and edges are well preserved for detecting small objects and for deriving masks with good edges in each RoI. The detail of the network is described in Section 2.

The paper is organized as follows: Section 2 illustrates the details of Mask R-CNN. In Section 3 experimental results are presented followed by discussion. Finally, conclusions and an outlook to future work are provided in Section 4.

2. MASK R-CNN

Mask R-CNN consists of three networks: feature pyramid network (FPN), regional proposal network (RPN) and fast R-CNN. The network design is shown in Figure 1. FPN consists of two pathways as shown in Figure 1. The bottom-up pathway is used to extract feature maps using ResNet (He et al., 2016). The output feature maps with different resolutions are from conv2, conv3, conv4, conv5 (C_2, C_3, C_4, C_5) with stride of {4, 8, 16, 32}. The high resolution feature maps is derived from low level features with less semantics, while the low resolution feature maps is derived from high level features with strong semantics. The top-down pathway is to merge high level feature maps with low level semantics feature maps from top to down iteratively. The pathway starts from putting the top feature map C_5 through a 1×1 convolution layer to create a feature map P_5 with 256-d. Then the merged map P_5 is upsampled by a scale of 2 and merged with the lower level feature map C_4 by element-wise addition after applying 1×1 convolution layer to reduce dimension of C_4 to 256-d. This process continues until 5 feature maps $\{P_2, P_3, P_4, P_5\}$ are derived, corresponding to $\{C_2, C_3, C_4, C_5\}$. Finally, a 3×3 convolution is applied on each merged feature maps to reduce aliasing effects due to upsampling. This merit of this approach is to assign semantics to each level of features.

In original RPN, a small network using two sibling fully connected layers to performing object versus non-object classification and bounding box regression using the feature map in the last conv layer. A 3×3 sliding window over the feature map is applied to extract 256-d vector to be fed into two fully connected layers. The reference bounding boxes, also called anchors, are generated through multiple scales and aspect ratios. The FPN is adopted with RPN by replacing the single feature map with feature pyramid maps. The multi-scale anchors are naturally obtained by assigning anchors of a single scale to each level. Five levels of feature maps $\{P_2, P_3, P_4, P_5, P_6\}$ are often used to include a wider range of anchor scales. P_6 is sub-sampled by stride of 2 of P_5. When an anchor is predicted, the corresponding optimal scale of feature maps is simply selected for the anchor. This mechanism is not only capable of extracting small objects, but also provides a optimal scale of feature map for further small object classifications and mask prediction with sharp edges.

Fast R-CNN uses the feature maps in each candidate boxes from RPN by RoIPool. RoIPool uses max pooling to aggregate any these region of interest (RoI) to a smaller 7×7 feature map. The smaller feature maps are vectorized and fed into two fully connected layers to predict the class and box offset. Mask R-CNN adopts Fast R-CNN and add a FCN network to predict mask in each RoI. However, RoIPool introduces a small misalignment of the smaller feature maps to the inputs, resulting in misalignment of final predicted mask. Mask R-CNN uses RoIAlign to align the aggregated feature maps to the inputs. Mask R-CNN defines a multi-task loss for each RoI to design a single stage training for three tasks: box classification, regression and mask segmentation, contributing to high computational efficiency in the training stage. The structure decouples box classification and mask segmentation. In each RoI, a binary segmentation is performed. Without competing with multi-classes in semantic segmentation, the binary segmentation also contributes to a good edges in the segmentation.

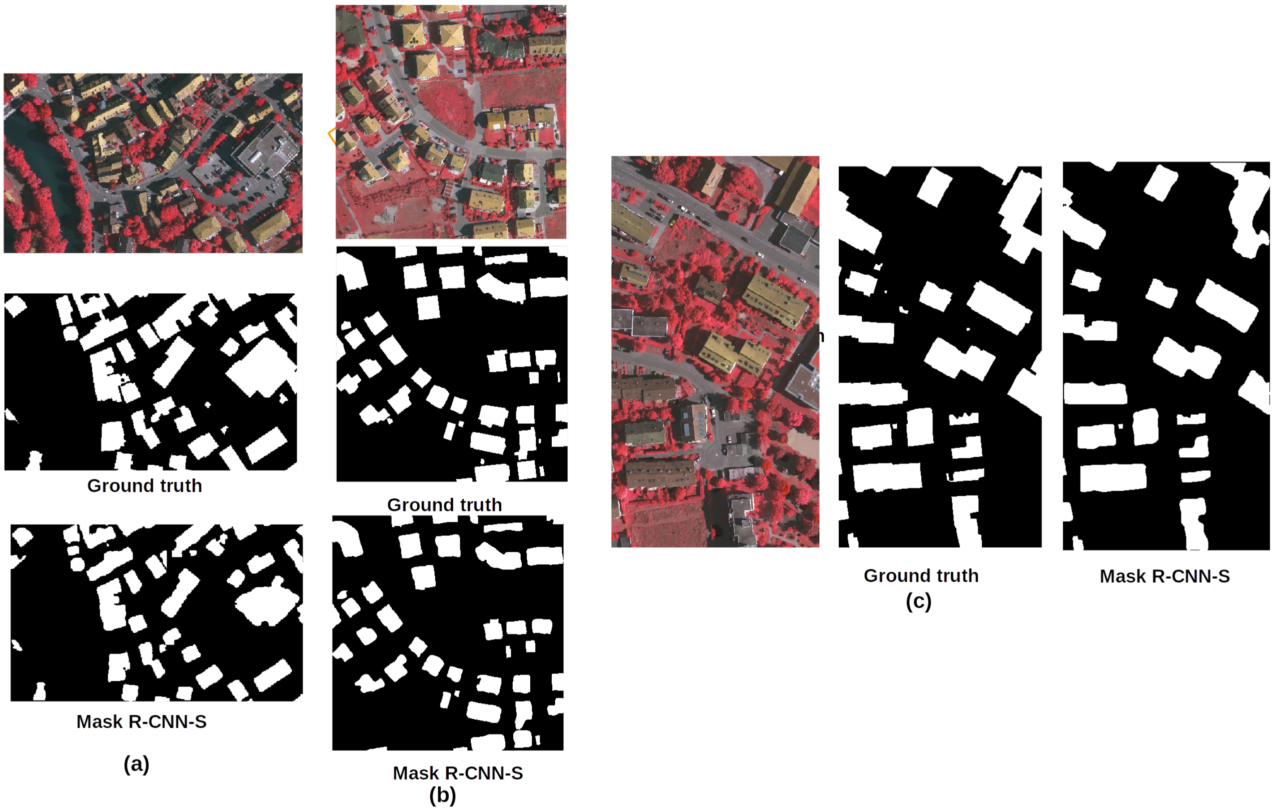

Figure 2. The building segmentation results of Mask R-CNN-S for test dataset with small anchor scale of $[16^2 - 256^2]$.

3. EXPERIMENT AND RESULT

3.1 Experiment

The experiment is performed on ISPRS benchmark Vaihingen data. Due to the relief displacement of buildings in the images, the DSMs interpolated from dense matching point clouds are used for correction of the distortion, especially for buildings. There are 33 areas for semantic segmentation contest, but only 16 images have ground truth labels. These 16 images are split for training, validation and test. The training set consists of 10 images (areas: 1, 3, 5, 7, 13, 15, 17, 21, 23, 28). The validation set consists of 3 images (areas: 28, 30, 32) and test set also consists of 3 images (areas: 26, 34, 37). In training samples, the smallest bounding box of objects is 21×21, while the largest bounding box is 598×477. All these images are split into 800×800. If there are parts left near image boundaries, these parts are merged to adjacent patches. The size is chosen to include one or more buildings and their contexture for extracting good features. If the patch size is chosen too large, the patch should be down-scaled in order to be handled by GPUs. The down-scaling approach will lose details in images. Finally, there are 59, 18 and 13 image patches for training, validation and testing.

Building segmentation program is written in python using TensorFlow and Keras libraries by customizing the Mask R-CNN program (Abdulla, 2017). One Tesla P100 graphics card with 12G RAM was used in this application. Two images with size of 1024×1024 fit to the RAM of the GPU as a mini-batch for training. Images of different sizes were rescaled to 1024×1024. As we wanted to detect buildings of different scales in the image, the choice of a good anchors scale was critical. When the images are rescaled, the smallest and largest bounding boxes becomes 27×27 and 765×610. As described in Section 2, the multi-scale anchors are defined by assigning a single scale to each level of feature map in the pyramid. In order to detect different scales of objects between 27×27 and 765×610, the natural choice is to set anchor scales to $32^2, 64^2, 128^2, 256^2, 512^2$ with three aspect ratios: $1:2$, $1:1$ and $2:1$. However, in such settings, many small boxes with anchor scale of 32^2 or its variants generated from RPN will be lost. As RPN generates tremendous amount of boxes, top-2000 ranked proposal boxes based on their prediction score are often selected for further class and mask prediction due to computational efficiency. The boxes with a small size are not easy to predict, resulting in low prediction scores. They are easy to be removed in this process. Therefore, an anchor scale of $16^2, 32^2, 64^2, 128^2, 256^2$ is chosen to make sure the boxes with areas of 32^2 can be preserved. However, the large objects may not be completely included in the largest box. Two experiments, Mask R-CNN-S with smaller scale $[16^2, 32^2, 64^2, 128^2, 256^2]$ and Mask R-CNN-L with larger scale $[16^2, 32^2, 64^2, 128^2, 256^2]$, are conducted. The result of Mask R-CNN-S is shown in Figure 2. The smaller anchor scale range made a trade-off in detecting small and large objects. Many small objects are detected, however, the segmentation within some big objects is not complete. The detailed comparison results with different anchor scales are shown in Section 3.3. The results in Figure 2 also show that the edges in the segmented buildings are sharp.

We trained the considered approach using one GPU with batch size of 2 for 90 epoches in Mask R-CNN. The pre-trained network on the COCO dataset is used for fine tuning. In the first 40 epoches, we trained the network with learning rate of 0.001, and in the last 50 epoches, we decreased the learning

Table 1. Quantitatively building segmentation result in ISPRS. (NOGT: number of objects (ground truth); NOCD: number of objects correctly detected; NOWD: number of objects wrongly detected). Mask R-CNN-L: Mask R-CNN with large anchor scale of $[32^2 - 512^2]$; Mask R-CNN-S: Mask R-CNN with small anchor scale of $[16^2 - 256^2]$.

Pixel-based	precision	recall	F_1	Object-based					
				NOGT	NOCD	NOWD	precision	recall	F_1
FCN	**0.843**	0.947	0.892	131	91	7	0.694	0.929	0.785
Mask R-CNN-L	0.833	**0.979**	0.900	131	106	4	0.809	0.964	0.869
Mask R-CNN-S	0.838	0.976	**0.901**	131	112	4	**0.855**	**0.966**	**0.896**

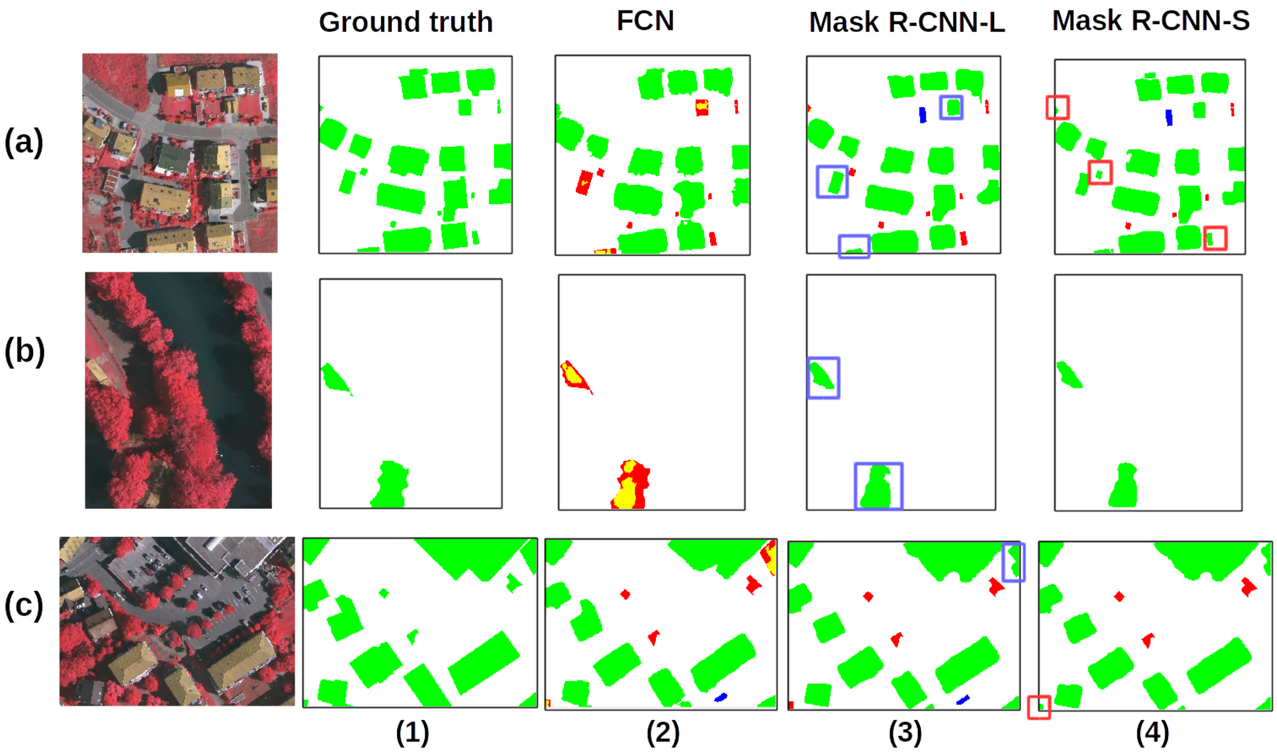

Figure 3. The results of the object-based building segmentation on three areas. Green: correctly detected objects. Red: miss-detected objects. Yellow: partially detected area in miss-detected objects. Blue: wrongly detected objects. Blue boxes in the result of Mask R-CNN-L show more correctly detected objects than FCN. Red boxes in result of Mask R-CNN-S show more correctly detected objects than Mask R-CNN-L.

.

rate to 0.0001. The weight decay of 0.001 and a momentum of 0.9 were set as the same with the original paper. Fully convolutional network was used for comparison with the Mask R-CNN in terms of segmentation on small objects and edges. FCN-8s was employed using skip layer mechanism to merge feature maps from layer: pool3, pool4 and conv7 in VGG16 (Simonyan , Zisserman, 2014). FCN-8s was tested with best performance on segmentation with details in the original paper. The pre-trained VGG-16 was used in FCN-8s. The same dataset was used for training, evaluation and testing. All images were also rescaled to 1024 × 1024 with a batch size of 2 for 100 epoches in FCN. The learning rate was set to 1e-5 according to (Chen et al., 2018). The same weight decay and momentum were defined as in the case of the Mask R-CNN. Mask R-CNN ran 7 hours for training, while FCN ran 2 hours for training.

3.2 Evaluation

Both pixel-based and object-based evaluation was performed. In both evaluation, three metrics are selected: precision, recall, F_1 score.

$$\text{precision} = \frac{TP}{GP}, \qquad \text{recall} = \frac{TP}{TP + FP}, \tag{1}$$

$$F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{2}$$

TP, FP and GP denote the true positives, false positives and positives of ground truth, respectively. Precision quantifies the correctness of building segmentation, while recall describes the completeness of building segmentation. The F_1 score is the harmonic average of the precision and recall. In pixel-based evaluation, the TP and FP are calculated based on intersection of building pixels between segmentation result and the ground truth. In object-based evaluation, the intersection between segmented building objects is calculated. Because the results from FCN segmentation are noisy, a connected component labelling is performed and the objects with less than 10 × 10 pixels are removed. TP is the number of objects correctly detected (NOCD). These objects are from the ground truth and have at least 60% area overlap with the detected objects. FP is number of object wrongly detected (NOWD). These objects are the detection which have at most 40% area overlap with the ground truth objects.

3.3 Result

3.3.1 Result on object-based evaluation In object-based evaluation in Table 1, FCN has the worst precision and recall with values of 0.679 and 0.929. Only 91 out of 134 buildings are detected and 7 buildings are wrongly detected. Mask R-CNN-S with anchor scale of $[16^2 - 256^2]$, and Mask R-CNN-L, with anchor scale of $[16^2 - 256^2]$, detected 15 and 21 buildings correctly, with 11% and 15% higher precision, respectively. Comparing the results of FCN and Mask R-CNN-L, all buildings found by FCN are found by Mask R-CNN-L and Mask R-CNN-S. An interesting finding is that all buildings detected by Mask R-CNN-L, shown in blue boxes in Figure 3, are partially detected by FCN, while Mask R-CNN-S finds more smaller buildings, shown in blue boxes in Figure 3, which are missed more by FCN. The skip layer structure in FCN do show its ability to use low level features to detect small objects partially, but FCN will perform worse when objects become smaller. With smaller anchor scale, Mask R-CNN-S shows better results in detecting very small objects. These detected small objects show relatively low detection confidence. This explains that if the smallest anchor scale is chosen as 32^2 in Mask R-CNN-L, these boxes generated from RPN are more likely to be ranked lower than 2000 due to their low confidences. They are removed before feeding to object classification and mask prediction. If the smallest scale is chosen as 16^2 in Mask R-CNN-S, these sizes of boxes are more likely to be removed instead of boxes with the size of 32^2. The FCN detects wrongly 3 more buildings compared to Mask R-CNN(-L and -S). The reason is also the skip layer structure does not manage to leverage low and high level features, while Mask R-CNN extract the optimal level of features for each objects for further object classification.

3.3.2 Result on pixel-based evaluation As shown in Table 1, FCN has around 1% higher precision in pixels, 0.843, than the Mask R-CNN with different anchor scales with precision of 0.833 and 0.838, but the recall, 0.947, is around 3% lower than two Mask R-CNN results. The lower recall of FCN shows that the the skip layer architecture produces more noisy results, while feature pyramid in Mask R-CNN improves correctness in building segmentation. As shown in Figure 4, the FCN produced more wrongly classified pixels (shown in blue), compared to Mask R-CNNs. The reason for little higher precision of FCN is that FCN has a bit better completeness in classifying large buildings, while the mask derived in R-CNN relies on object detection. Large anchor scales [32-512] have tendency to miss small buildings which has been discussed in Section 3.3.1, while small anchor scale [16-256] have tendency to detect less pixels in large buildings. As shown in blue boxes in Figure 4, the pixels segmented in large building in Mask R-CNN with a small anchor scale are a bit worse than Mask R-CNN with a large anchor scale. Due to the trade-off in segmenting small and large objects, Mask R-CNN with different anchor scales of Mask R-CNN do not show significant differences in pixels. When comparing the results for edge regions, FCN has much more noisy results than Mask R-CNNs as shown in the yellow boxes in Figure 4. The result matches with worse small objects detection discussed in Section 3.3.1, the skip layer in FCN loses detailed information, such as small objects and edges.

4. CONCLUSION

In this paper, we employed Mask R-CNN to solve two problems in building segmentation in airborne VHR images: detecting buildings in different scales and segmenting accurately building edges. Standard CNNs have a tendency to lose low level features due to pooling or stride. Skip layer structure merges the low and high level features for prediction, however, different scales of objects can hardly be classified using the same scale of features. Mask R-CNN starts from feature pyramid network creating different scales of semantically rich features. The feature pyramids are used by the region proposal network to generated objects with various scales with the corresponding optimal scale of features. The features with high and low levels of information are used for better object classification on small objects and mask prediction on edges. The results on ISPRS benchmark dataset show that Mask R-CNN outperforms FCN with around 15% in detecting objects, especially small objects. The significant difference between FCN and Mask R-CNN is not in segmenting pixels, however, when comparing the results in edge regions, Mask R-CNN produces much better results. The another important finding in our paper is that anchor scale in Mask R-CNN is a critical factor that influences the results of segmentation of objects on different scales. This paper also gives an insight on how to choose the good anchor scale for specific type of data.

In further work, there are two posible extensions to improve the building segmentation on small objects and edges: (1) the DSM can be added in the same framework as several small buildings are hardly to be distinguished using color information. (2) More accurate training samples should be provided. The training samples from ortho-rectified VHR images using DSMs from dense matching. However, the dense matching suffers from shadows and low textures. The edges in many building are severely distorted. In (Zhou et al., 2018), an old LiDAR data is used to guide dense matching while detecting the changes between LiDAR data and images. In this approach, the training samples with better quality can be used to improve the quality of building segmentation.

ACKNOWLEDGEMENTS

The Vaihingen data set was provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF) (Cramer, 2010) http://www.ifp.uni-stuttgart.de/dgpf/DKEP-Allg.html. This work was carried out on the Dutch national e-infrastructure with the support of SURF Cooperative. This work is part of the TTW Maps4Society project generation of up-to-date 3D city models for water management, which are (partly) financed by the Netherlands Organization for Scientific Research (NWO).

REFERENCES

Abdulla, Waleed, 2017. Mask r-cnn for object detection and instance segmentation on keras and tensorflow. `https://github.com/matterport/Mask_RCNN`.

Bittner, Ksenia, Adam, Fathalrahman, Cui, Shiyong, Körner, Marco, Reinartz, Peter, 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, 2615–2629.

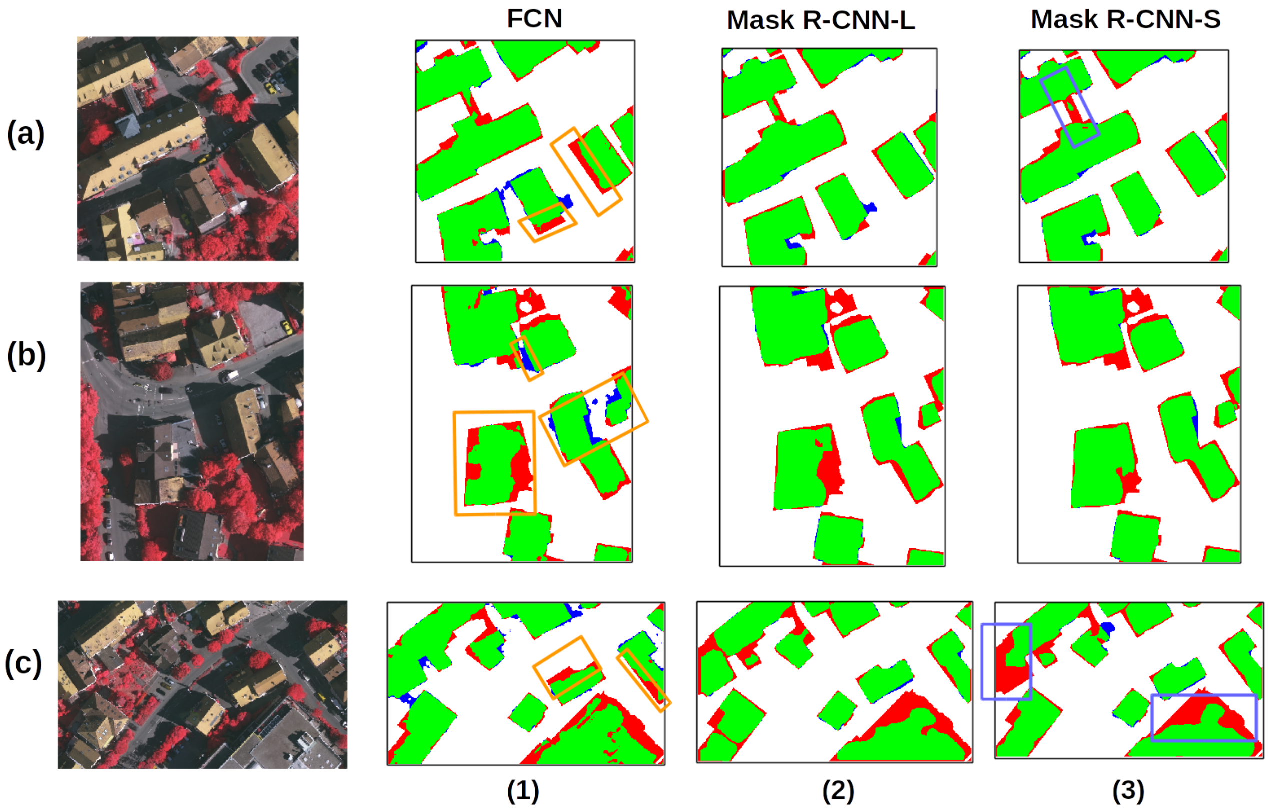

Figure 4. The pixel-based evaluation of building segmentation results on three areas. Green: correctly detected building pixels. Red: miss-detected building pixels. Blue: wrongly detected building pixels. Blue boxes in the result of Mask R-CNN-S show the miss-detected pixels due to small anchor scales selected. Yellow boxes in the result of FCN show more wrongly segmented pixels in building edges than Mask R-CNN.

.

Blaschke, Thomas, 2010. Object based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65, 2–16.

Chen, Y, Gao, W, Widyaningrum, E, Zheng, M, Zhou, K, 2018. Building classification of VHR airborne stereo images using fully convolutional networks and free training samples. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 42.

Cramer, Michael, 2010. The DGPF-test on digital airborne camera evaluation–overview and test design. *Photogrammetrie-Fernerkundung-Geoinformation*, 2010, 73–82.

Girshick, Ross, 2015. Fast r-cnn. *Proceedings of the IEEE International Conference on Computer Vision*, 1440–1448.

He, Kaiming, Gkioxari, Georgia, Dollár, Piotr, Girshick, Ross, 2017. Mask r-cnn. *Computer Vision (ICCV), 2017 IEEE International Conference on*, IEEE, 2980–2988.

He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, Sun, Jian, 2016. Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Kaiser, Pascal, Wegner, Jan Dirk, Lucchi, Aurélien, Jaggi, Martin, Hofmann, Thomas, Schindler, Konrad, 2017. Learning aerial image segmentation from online maps. *IEEE Transactions on Geoscience and Remote Sensing*, 55, 6054–6068.

Lin, Tsung-Yi, Dollár, Piotr, Girshick, Ross, He, Kaiming, Hariharan, Bharath, Belongie, Serge, 2017. Feature pyramid networks for object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2117–2125.

Long, Jonathan, Shelhamer, Evan, Darrell, Trevor, 2015. Fully convolutional networks for semantic segmentation. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 3431–3440.

Maggiori, Emmanuel, Tarabalka, Yuliya, Charpiat, Guillaume, Alliez, Pierre, 2017. Convolutional neural networks for large-scale remote-sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55, 645–657.

Marmanis, Dimitrios, Schindler, Konrad, Wegner, Jan Dirk, Galliani, Silvano, Datcu, Mihai, Stilla, Uwe, 2018. Classification with an edge: Improving semantic image segmentation with boundary detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 135, 158–172.

Ren, Shaoqing, He, Kaiming, Girshick, Ross, Sun, Jian, 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 91–99.

Ren, Yun, Zhu, Changren, Xiao, Shunping, 2018. Small Object Detection in Optical Remote Sensing Images via Modified Faster R-CNN. *Applied Sciences*, 8, 813.

Ronneberger, Olaf, Fischer, Philipp, Brox, Thomas, 2015. U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical Image Computing and Computer-assisted Intervention*, Springer, 234–241.

Simonyan, Karen, Zisserman, Andrew, 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Wang, Shenlong, Bai, Min, Mattyus, Gellert, Chu, Hang, Luo, Wenjie, Yang, Bin, Liang, Justin, Cheverie, Joel, Fidler, Sanja, Urtasun, Raquel, 2017. Torontocity: Seeing the world with a million eyes. *2017 IEEE International Conference on Computer Vision (ICCV)*, IEEE, 3028–3036.

Yuan, Jiangye, 2018. Learning building extraction in aerial scenes with convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40, 2793–2798.

Zhou, Kaixuan, Gorte, Ben, Lindenbergh, Roderik, Widyaningrum, Elyta, 2018. 3D building changedetection between current VHR images and past LIDAR data. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 2.