SEMANTIC SEGMENTATION OF BUILDING IN AIRBORNE IMAGES

S. Huang, F. Nex, Y. Lin, M.Y. Yang

ITC Faculty of Geo-Information Science and Earth Observation, University of Twente, The Netherlands
{s.huang, f.nex, y.lin-1, michael.yang}@utwente.nl

Commission II, WG II/4

KEY WORDS: Buildings, Semantic Segmentation, Deep learning, 3D features

ABSTRACT:
Building is a key component to the reconstructing of LoD3 city modelling. Compared to terrestrial view, airborne datasets have more occlusions at street level but can cover larger area in the urban areas. With the popularity of the Deep Learning, many tasks in the field of computer vision can be solved in easier and efficiency way. In this paper, we propose a method to apply deep neural networks to building façade segmentation. In particular, the FC-DenseNet and the DeepLabV3+ algorithms are used to segment the building from airborne images and get semantic information such as, wall, roof, balcony and opening area. The patch-wise segmentation is used in the training and testing process in order to get information at pixel level. Different typologies of input have been considered: beside the conventional 2D information (i.e. RGB image), we combined 2D information with 3D features extracted from dense image matching point clouds to improve the performance of the segmentation. Results show that FC-DenseNet trained with 2D and 3D features achieves the best result, IoU up to 64.41%, it increases 5.13% compared to the result of the same model trained without 3D features.

1 Introduction

Due to the explosion of urbanization and the increase in population in recent years, a new challenges has to be faced in regard to the planning and environmental sustainability or urban areas. To tackle these problems, the use of more detailed and complete geographic information is mandatory. “Smart Cities” aim at delivering smart and complete information thanks to digital technologies. In this regard, the realization of 3D city modeling allows to interoperate and share many data in an efficient way. Different levels of city models can be then generated (Dimopoulou, et al., 2014). City Geography Markup Language (CityGML) is considered the standard for 3D city modeling. In CityGML, building parts and accessories can be classified into four levels-of-detail from LoD1 to LoD4 (Gröger & Plümer, 2012). In LoD1, buildings are modelled in a generalized way, like blocks. In LoD2, the roof shape of the building is represented. LoD3 is a more detailed level, openings (window, door) and detailed rood structures (chimney) are added for buildings, and in LoD4, the interior (room) are represented too. Currently, the low level (LoD1 and LoD2) can be generated (almost) automatically, but this process is not feasible for LoD3. Many details such as the building components cannot be reliably extracted in an automated way and therefore they cannot be automatically inserted into a 3D model.

The semantic segmentation of a building can be considered as ais sub-problem of the automatic generation of virtual cities with LOD3 models. The task of the building façade segmentation is to assign each pixel of human-made structures to a semantic label such as window, balcony and door. However, manual delineation over large urban areas is time-consuming. An automatic way for semantic segmentation of building is the unique choice from a practical point of view.

Early methods for building façade segmentation were based on an appropriate shape grammar (Gadde et al., 2018) following the predefined architectural constraints (e.g. windows are of the same size on the façade and not placed randomly; doors can be found on the first floor at street -level; the roof is above the top floor; all balconies have the same dimensions, etc.). These rules can reduce the errors of the segmentation result but they heavily rely on the prior knowledge.

Machine learning is an efficient and automated method to parse building. There are a few classifiers that can be applied to tackle this task, for example, Support vector machine (SVM), RANSAC (Boulaassal, et al., 2007), randomized decision forest (Yang & Forstner, 2011). However, these algorithms typically return noisy pixels in their segmentation results, due to the lack of neighboring information (Rahmani, et al., 2017). Conditional Random Field (CRF) (Lafferty, et al., 2001) is also a popular method to refine the output of the classifier to improve the accuracy of the result.

Recently, deep learning outperformed the traditional method (SVM, RF) in terms of accuracy and robustness. Convolutional Neural Networks (CNNs) have shown a good performance and high efficiency in image recognition, object detection, semantic segmentation. (Long, et al., 2015) proposed an end-to-end network using fully convolutional architecture-FCN outperforming previous algorithms in the task of semantic segmentation. Compared to the classical convolutional neural networks, FCN replaces the final fully connected layer with a convolutional layer and outputs a pixel-wise labeled image instead of a classification score. FCN accepts arbitrarily sized images as the input and recovers shrunken images after a series of convolutional layers thanks to the deconvolutional layer (Garcia-Garcia, et al., 2017). However, training models from the beginning is a time-consuming work and cannot produce good results with random initialization. Thus a common trend in the segmentation is to apply transfer learning (Yosinski et al., 2014) fine-tuning the pre-trained classification networks.

Building data can be captured from multiple platforms. Compared to the terrestrial data, the airborne oblique imagery is more productive in the urban area as it can cover larger areas and it can acquire the same object from different images. Compared to aerial nadir images many more details can be then acquired and used to the further improve the generation of 3D models (Xiao, et al., 2012). The cost of oblique images is also lower than other terrestrial methods.

In this paper the use of FCN for façade segmentation is investigated. In particular, two Deep Neural Networks, namely

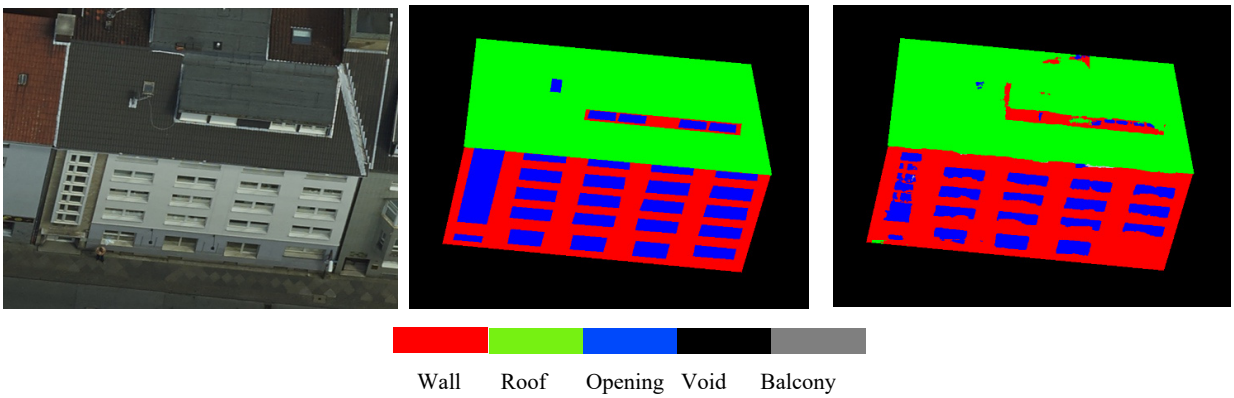

Figure 1: Examples of our task, from left to right are Original image, Ground truth, Result from DenseNet trained with 2D and 3D feature.

FC-DenseNet and DeepLabV3+ are adopted to parse buildings from oblique images captured by airborne systems.

The contribution of the study is that the input of the network includes not only 2D image information (RGB), but also point clouds to provide extra 3D information (the third component of the normal vector) for improving accuracy. For the training process, weighted loss function is used to solve the problem of imbalanced classes. We also use patch-wise segmentation in our task to reconstruct the original image sizes and we choose the maximum probability in the score map instead of their direct combination to delineate the negative effect when reconstructing from small patches to original images.
.

2 Related work

Many researches have been working on the façade detection and classification. The aim of these works has been to estimate the position and size of various structural (e.g., window, door, roof) and non-structural elements (e.g., sky, road, building) exploiting their shape or their appearance on the given images (Fröhlich, et al., 2010). The previous works can be classified into different categories according to the data source: image-based (2D) and laser-based (3D) algorithms. These can be then subdivided into the airborne and the terrestrial according to the used platforms.
(Cohen et al., 2014) presented a method using dynamic programming algorithm to praise the façade of the building and applying the hard-architectural constraints. Gadde et al., 2015) have used the learning split grammars from annotated images to perform the pixel-wise classification. In (Delmerico et al., 2011) a method has been proposed using three main steps: discriminative modelling, candidate plane detection through PCA and RANSAC, and energy minimization of MRF potentials, refining the result with the plane fitting. Martinović et al., 2012 shows a three layer architecture where the low-level information is given by the semantic segmentation, middle-level is based on a pairwise multi-label Markov Random Field (MRF) solved by a graph-cut algorithm about objects in the facade, and top-level is according to the architectural knowledge. Randomized decision forest (RDF) is also a good classifier to classify the building façades. (Yang & Wolfgang, 2011) demonstrated an approach of region-wise classification by RDF and local features refining the result with the conditional random field (CRF). They trained a RDF on the labelled data and split them by a decision tree learning algorithm. (K. Rahmani et al., 2017) proposed a method using a Structure Random Forest for façade labelling and get a good performance result on the ECP and Graz façade datasets. Fully connected CRFs can model long-range spatial dependencies and make use of contextual information. Li & Yang, 2016 used the fully connected CRF (all nodes are connected in pairs) as the basic framework for the façade parsing task. They chose the trained Textonboost as the unary classifier and obtained maximum posterior marginal (MPM) results by filter-based mean-field approximation inference. The use of oblique images is a way to capture multi-views of building facades. In this regard, Tu et al., 2017 extract the feature following local symmetrical and using a sliding window to determine the location of the local symmetry feature point. Xiao et al., 2012 make use of the oblique imagery with large tilt angle to solve the problem of occlusions and get more detailed information about the façade.
Recent years, deep learning becomes a popular method in the field of computer vision. It has proven to have a good performance for tasks like object detection, classification, and segmentation. Many remote-sensing applications can be also achieved using deep learning, such as hyperspectral image analysis, interpretation of SAR images, interpretation of high-resolution satellite images, multimodal data fusion, and 3D reconstruction (Zhu et al., 2017). Kujtim Rahmani & Mayer, 2018 mainly introduced the Region Proposal Network (RPN) based on a Convolutional neural network to generate the prior information for the building elements, such as window, door, balcony with their probability, and then put it into the Structured Random Forest as the input.
(Long et al., 2015) proposed the first Fully Convolutional Networks (FCN) that is an end-to-end deep neural network for semantic segmentation (Figure 2). It makes dense predictions for semantic segmentation using arbitrary size of the input by adding up-sampling layers to restore the spatial resolution of the input. A skip connection is also added to the networks. A CNN can be converted into FCN by replacing the fully connected layers with a 1 x1 size of convolutional layer. Therefore, the existing models of CNNs can also be used into FCN. (Liu et al., 2017) applied FCN into the 2D façade paring problem, they proposed a symmetric regularization term and to train the neural network with a novel loss function and boosting the performance with the post-processing based on object detection. (L. C. Chen et al., 2018) proposed an idea combined with the deep convolutional neural networks based on ResNet and fully-connected conditional random fields.

3 Methodology

The developed methodology can be divided in a sequence of steps described in the following sub-sections.

3.1 3D feature extraction

The third component of the normal vector is the 3D feature involved in convolutional neural networks. This can tell whether surfaces are horizontal, vertical or slanted. The normal vector is derived from a cluster of neighboring points which can be selected by different searching strategies and searching ranges. Our work uses 'K-nearest neighbors' as the searching strategy and pick 100 neighboring points to calculate the normal vector for each point.

3.2 Feature combination

Our networks are based on 2D CNN architectures: the 3D feature is therefore projected into image space and taken as the fourth channel of the network input. The projection to oblique airborne images is based on P-matrices which are obtained after dense matching point cloud generation in the Pix4D software. During the projection, one point can be associated with image patches of different sizes: pixels within the same patch share the same 3D feature. When multiple points are projected to the same patch, the averaged feature value is assigned to the patch. In real experiments, small patches leave voids in image space, while large patches reduce the void percentage but, at the same time, lead to coarse features that are insufficient to provide detailed information. Half resolution point clouds are used in our experiments. To keep the balance between void percentage and details of information, the optimal patch size is set as 4 pixels by 4 pixels.

3.3 Patch-wise segmentation

In our task, our data have different resolutions. Due to the limited memory of the GPU and effiency the patch-wise strategy has been adopted to train our neural network. Compared to image resizing, patch-wise segmentation can keep the contextual information and keep the original shapes of images, without any distortion. First, in the training process, the images will be split y small patches (320×320). To get a better performance of the border, we take 50% size of each patch as the overlapping region to deal with the gap between adjacent images.

In the testing stage, the original images are then reconstructed from small patches using a fusion strategy. The neural networks give a proper probability distribution for each pixel by $softmax$ function. Where, $\boldsymbol{e}^{y_i}$ is the scores of the input in the form of one-hot encode, i with the length equal to the number of classes J . In this task J equals to 4, i=0,1,2,3.

$$S_i = \frac{e^{y_i}}{\sum_j e^{y_j}} \quad (1)$$

In the overlapping region, the probability of each pixel chooses the maximum probability between two regions, s_1 and s_2 .

$$P(i) = argmax(s_1, s_2) \quad (2)$$

3.4 Imbalance class

Classes with fewer pixels are likely to cause the imbalance problem during the training. In this paper, the weighted loss function has been used to solve the imbalanced training.

Cross entropy loss is a common loss function that can be used in segmentation tasks. Where y is the ground truth in the form of one-hot encode and $\hat{y}$ refers to the prediction generated by the last layer output. The length of K equals to the number of labels (1,2, 3…, k).

$$\mathrm{L} = -\frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=0}^{K-1}(y_{ik}) \cdot \log(\hat{y}_{ik}) \quad (3)$$

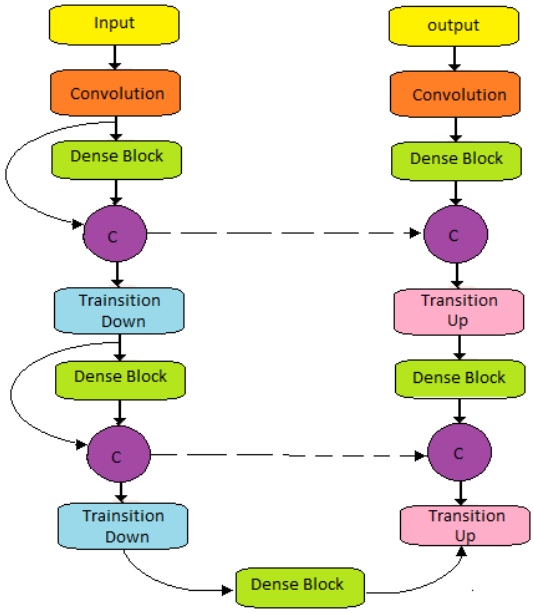

Figure 2: The structure of FC-DenseNets from (Jegou et al., 2017).

The weighted loss function is shown in following, where W_c is the class weight computed by the number of pixels for each class in training images. L is the cross-entropy loss.

$$L_{weighted} = L \cdot W_c \quad (4)$$

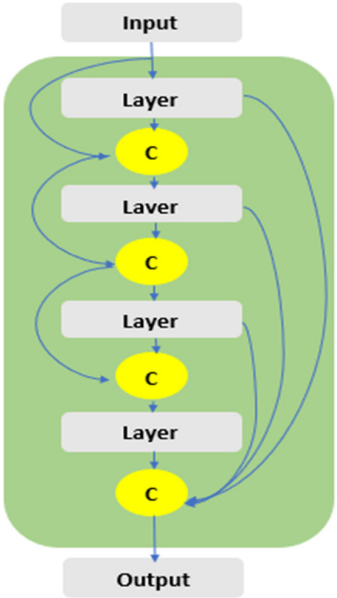

Figure 3: Dense Block from (Jegou et al., 2017).

3.5 Networks

There are two main structures of models used for semantic segmentation tasks in deep neural networks, namely spatial pyramid pooling and encoder-decoder structure. The major advantage of the first one is its capability to capture multi-scale

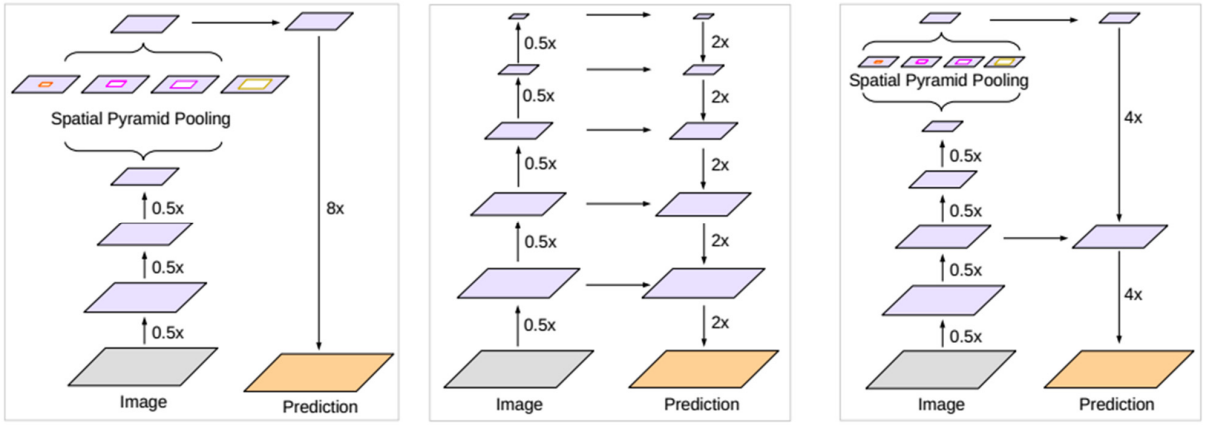

Figure 4: From left to right a) Spatial pyramid structure, b) Encoder-Decoder structure, c) Encoder-Decoder with Atrous convolution. (Chen et al., 2018).

information. In the following sub-sections, the network architectures that we used in our task are introduced.

3.5.1 FC-DenseNets

Fully Convolutional DenseNets was proposed by (Jegou et al., 2017), as shown in Figure 2. It is based on DenseNets (Huang et al., 2017) and extended to deal with the problem of semantic segmentation task by combining FCN with DenseNets. The goal is to do the classification, also achieve the pixel-to-pixel segmentation and keep the original image resolution at the same time by adding the up-sampling path of the FCN. This network contains less parameters and is not necessary to be pretrained on large datasets.

The main feature of DenseNets is given by the use of Dense Blocks. Figure 3 shows a Dense Block of 4 layers. Starting from an input x_0 with m feature maps, after going through the first layer, the output x_1 of dimension k is generated by applying $H_1(x_0)$. The input of the next layer is from stacked features by a concatenation $([x_0, x_1])$ (Jegou et al., 2017).

3.5.2 DeepLabv3 plus

The main advantage of DeepLabv3 is that it can capture the contextual information at multiple scales by applying a spatial pyramid pooling module. However, there is a certain drawback associated with the boundary of objects. Deeplabv3 plus, as shown in Figure 4, improved the performance based on DeepLabv3 (see the next section), by adding an encoder-decoder structure that is able to obtain sharp object boundaries (Chen et al., 2018). The Xception model had provided promising images classification results. In DeepLabv3 plus, the author modified the model and adapted it to semantic segmentation tasks, as the new backbone to extract features. In the performed tests, ResNet101 has been used as backbone as used in DeepLabv3. Figure shows the three different structures to capture multi-scale context.

ASPP – Atrous Spatial Pyramid Pooling
Atrous Spatial Pyramid Pooling was applied in DeepLabv3 to capture contextual information in different scales with different rates. It can compensate for the loss of information by using pooling or convolution layers. In this regard, the conventional stride pooling, although the main features are kept in the final feature map, it loses the small detailed information, like boundaries.

4 Experiments

4.1 Airborne datasets

The data is based on (Lin, et al. 2018) captured from Dortmund city center on July 7th 2016, shown in Figure 5. The dataset consists of multiple views images. The ground sampling distance is 4.5cm for the oblique image. In this paper, 4 classes are considered, roof, wall, balcony and opening (window and door). The example of the ground truth in our task is shown in Figure 6.

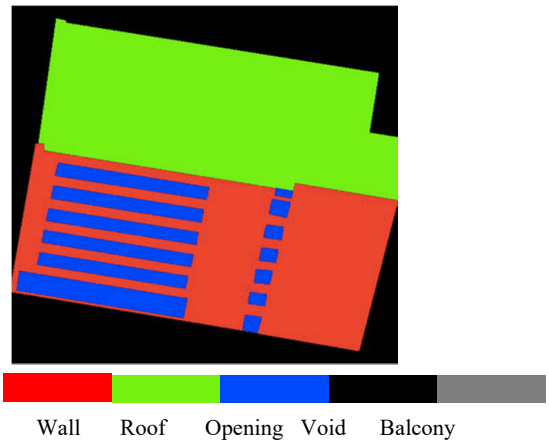

Figure 6: An example of the ground truth.

4.2 Region of Interest

The facades in each original image are in different scales. Then, the facades are not rectified into vertical perspectivse, so there will be some noise produced by the area surrounded the facade. To improve the performance of result and train it in an efficiency way, a smaller region of interest in correspondence of the façade is used as the input of networks before splitting it into patches. An example is shown in the Figure 7.

Figure 5: Dense matching point cloud of study area.

Figure 7: Region of Interest Selection.

4.3 Experiment setup

Due to the limited data, the data augmentation is used in our task to increase the training data. The number of original training data is 160 images. After splitting into patches, there are 1132 images used for training. The framework of the networks is based on (Seif, 2018). As already mentioned, ResNet is used as the back-bone to extract features, using the ImageNet pretrained model. Then FC-DenseNet and DeepLabV3+ are fine-tuned on our dataset.

4.3.1 2D segmentation

The input of the networks in 2D segmentation only contains 3 channels (RGB). Several tests have been performed to define the most proper configuration. In this regard, the data augmentation has been adopted too. The horizontal flip has been used, reversing the images horizontally. Rotation values were set to 20 degree $[-20\ 20]$, randomly changing this value for each patch. Brightness value was set to 0.2: this value refers to randomly change the factor of brightness from 0 to 20%.

Data augmentation	Value
Horizontal flip	True - horizontal
Rotation	20
Brightness	0.2

Table 1: Data augmentation in training process

The learning rate was set to 0.0001 in our task, and decay rate set was 0.995 of original learning rate after each epoch to avoid overfitting. Due to the limited memory of the resource small batches are used in our training process. For Deep LabV3+, Batch size is set less than 12, so we did not fine-tune the batch normalization.

Parameter	Value
Learning rate	0.0001
Decay rate	0.995
Batch size	4 (DenseNet) / 8 (Deeplab)
Epochs	80

Table 2: Parameters in training process

4.3.2 Combination of 2D and 3D features

The input of the networks in 2D was combined with 3D feature in 4 channels, RGB and normal vector. The weights pretrained on ImageNet used the initial 3 channels (RGB), while the extra channel for normal vector was initialized with 0 values. The first layer was modified into $3 \times 3 \times 4 \times 64$ instead of $3 \times 3 \times 3 \times 64$ to fit the 4 channels input. Comparing to the 2D segmentation, the training strategy almost the same, but removes rotation from the data augmentation and set epochs to be 100.

4.4 Accuracy assessment

To evaluate the performance of the segmentation result, there are some metrics to be defined. Overall accuracy is the metric evaluated for the whole image. IoU is a common way in dense prediction tasks. We take the mean over the IoU of each predefined class. In these equations, TP refers to true positives. while FP refers to the false positive, FN refers to false negatives and TN indicates the true negatives. The formulas of the used metrics are reported below.

$$Accuracy_{overall} = \frac{TP}{TP+FN} \quad (5)$$

$$IoU = \frac{TP}{TP+FP+FN} \quad (6)$$

4.5 Results and discussion

42 complete images were used for testing before splitting into patches. Some visualizations of results is shown in Figure 8. The first row are the original images, while the second gives the ground truth and the third shows the result from FC-DenseNet trained with only 2D information. The fourth row shows the result from DeepLabV3+ trained using only 2D information, the

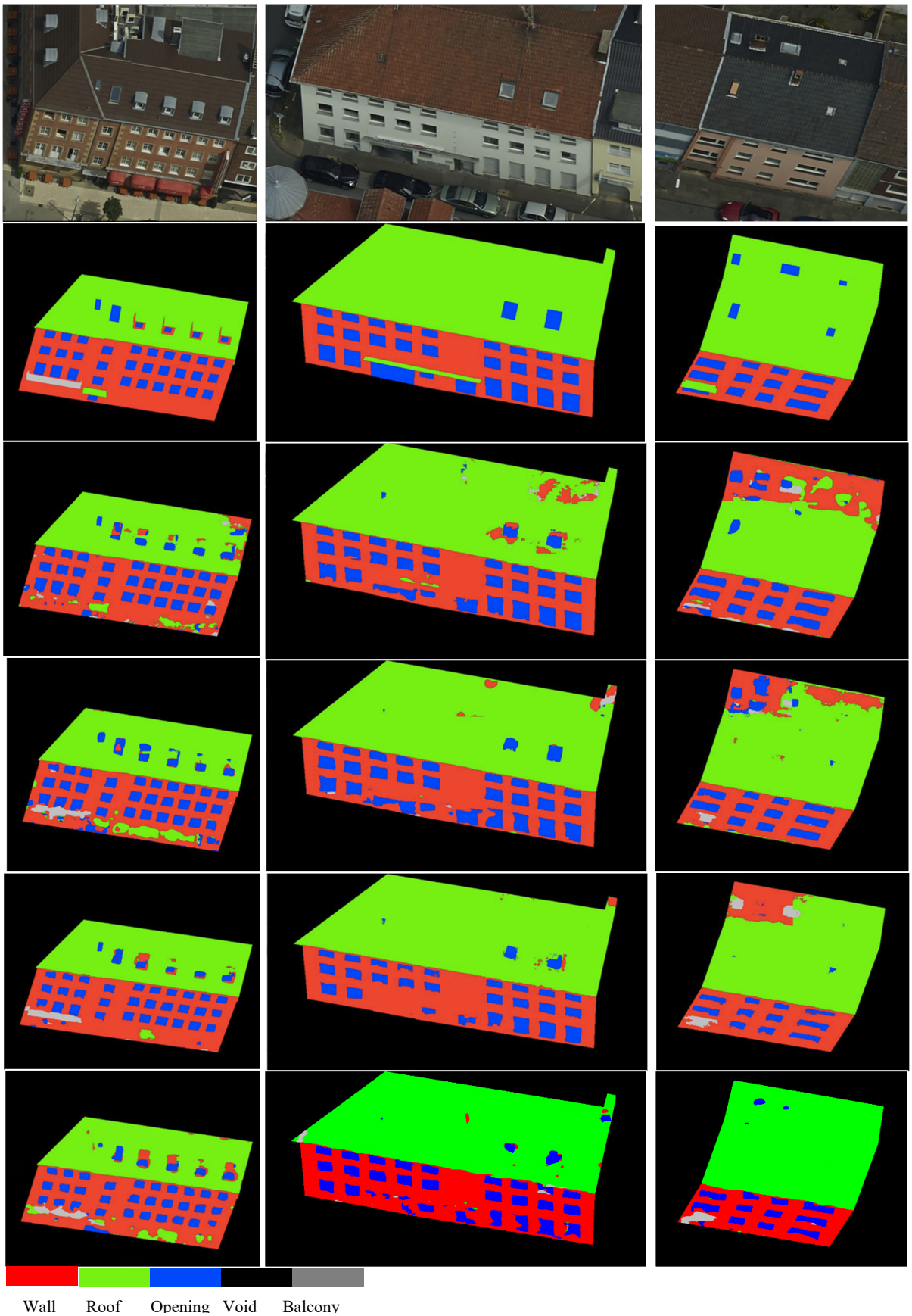

Figure 8: First row: Original images. Second row: Ground truth. Third row: Results generated by FC-DenseNet. Fourth row: Results generated by DeepLab. Fifth row: Results generated by FC-DenseNet 2D with 3D features. Sixth row: Results generated by DeepLab 2D with 3D features.

fifth row shows the result from FC-DenseNet trained with 2D and 3D feature, while the sixth row provides the result from DeepLabV3+ trained with 2D and 3D features.

From the Table 3, it can be seen that the bet result is obtained by FC-DenseNet trained with 2D and 3D features: it achieves 64.41% IoU and 91.30% accuracy. The second best is given by DeepLab V3+ trained with 2D and 3D features and it obtains 62.16% IoU and 91.10% accuracy.

Compared to the only use of 2D information, the overall accuracy and the IoU using 2D and 3D features increased for both DeepLabV3+ (from 89.08% to 91.10%) and for FC-DenseNet, (from 88.42% to 91.30%). In addition, the IoU of FC-DenseNet improves more than 5%, while DeepLabV3+ achieves a less extensive improvement (1% less improvement).

Results on roof in FC-DenseNet and DeepLabV3+ using all the features are better than these two models only trained with 2D information: the overall accuracy increased from 91.76% to 94.61% and 92.25% to 95.78% respectively.

In correspondence of walls, the model trained with 2D and 3D feature worsens the model only using 2D information. The classification of the openings using FC-DenseNet trained using 2D and 3D feature achieved the best performance (from 78.1% to 92.89%). On the other hand, DeepLabV3+ gave opposite results using the same typologies of features, decreasing from 81.25% to 65.42%. Results of balcony in FC-DenseNet get a little improvement from 78.89% to 79.18% by adding 3D feature, while the same configuration in DeepLabV3+ decreases 7.65% than using the only 2D information. Mean accuracy of class in FC-DenseNet improves 0.29% while decreases 4.99% in DeepLabV3+.

Class	2D-FC	2D-DL	FC-23	DL-23
Roof	91.76	92.25	94.61	**95.78**
Wall	**83.84**	82.12	64.30	82.08
Opening	78.10	81.25	**92.89**	65.42
Balcony	61.66	**70.73**	64.93	63.08

Table 3: Results from four tests. 2D-FC refers to FC-DenseNet used only 2D information. 2D-DL refers to DeepLabV3+ used only 2D information. FC-23 represents FC-DenseNet used 2D and 3D information. DL-23 represents DeepLabV3+ used 2D and 3D information.

	2D-FC	2D-DL	FC-23	DL-23
IoU	59.28	62.09	**64.41**	62.16

Table 4: mean IoU for different models.

Overall, the achieved results indicate that model predictions get benefits from 3D information and achieve the best performance. With RGB input, there are some confusions between roof and wall. Adding 3D features, as it can be seen from Figure 8, reduces the misclassified pixels: the normal vector can help the network to easily distinguish between these two classes and to solve the confusion by adding extra vertical spatial information to. On the other hand, the normal vector has limited effects on classifying other classes where the geometric information provided by 3D data is less discriminative in the classification process: in this case, the performance in the classification of opening and balconies can slightly decrease. These trends have been confirmed in both networks. The results provided by architectures provide similar results in terms of accuracy.

5 Conclusion and further work

In this paper, we applied two neural networks, FC-DenseNet and DeepLabV3+ to segment buildings captured from the airborne oblique camera system. Four classes have been considered in our task: roof, wall, opening area, and balcony. Instead of traditional terrestrial data, airborne data cover larger area to be investigated such as roofs that are difficult to be covered in terrestrial datasets. In our task, we not only consider conventional 2D information, but also the training data combined the 2D information with a 3D feature (the third component of the normal vector) into the networks. The results indicate that 3D features can correct misclassified pixels by providing extra spatial information, such as confusions between wall and roof. The IoU and accuracy of in FC-DenseNet and DeepLabV3+ all increase, and FC-DenseNet trained with 2D combined 3D feature get the best result with 91.30% accuracy and 64.41% IoU.

In the further work, other 3D features can be involved in the training process to provide more spatial information to improve the performance of other classes. Also, CRF can be applied as the post-processing to refine the result. Furthermore, the limitation of this work is the limited memory of resource, we can implement more advanced neural networks and test large resolution of images on the semantic segmentation of buildings.

6 References

Boulaassal, H., Landes, P., Grussenmeyer, F., & Tarsha-Kurdi, F. (2007). Automatic Segmentation of Building Facades using Terrestrial Laser Data. ISPRS Workshop on Laser Scanning 2007 and SilviLaser 2007, XXXVI, 65–70.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation.

Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2018). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(4), 834–848.

Cohen, A., Schwing, A. G., & Pollefeys, M. (2014). Efficient structured parsing of facades using dynamic programming. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition.

Delmerico, J. a., David, P., & Corso, J. J. (2011). Building facade detection, segmentation, and parameter estimation for mobile robot localization and guidance. 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, 1632–1639.

Dimopoulou, E., Tsiliakou, E., Kosti, V., & Floros, G. (2014). Investigating integration possibilityies between 3d modeling techniques. In In Proceedings of 9th International 3D GeoInfo Conference (pp. 1–16).

Floros, G., Pispidikis, I., & Dimopoulou, E. (2017). Investigating integration capabilities between IFC and citygml lod3 for 3D city

modelling. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives, 42(4W7), 1–6.

Fröhlich, B., Rodner, E., & Denzler, J. (2010). A fast approach for pixelwise labeling of facade images. Proceedings - International Conference on Pattern Recognition, 3029–3032.

Gadde, R., Jampani, V., Marlet, R., & Gehler, P. V. (2018). Efficient 2D and 3D Facade Segmentation Using Auto-Context. IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(5), 1273–1280.

Gadde, R., Marlet, R., & Paragios, N. (2015). Learning Grammars for Architecture-Specific Facade Parsing. International Journal of Computer Vision, 117(3), 290.

Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., & Garcia-Rodriguez, J. (2017). A Review on Deep Learning Techniques Applied to Semantic Segmentation. In arXiv preprint (pp. 1–23).

Gröger, G., & Plümer, L. (2012). CityGML – Interoperable semantic 3D city models. ISPRS Journal of Photogrammetry and Remote Sensing, 71, 12–33.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In IEEE Conference on Computer Vision and Pattern Recognition, pp. 2261–2269.

Jegou, S., Drozdzal, M., Vazquez, D., Romero, A., & Bengio, Y. (2017). The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 1175–1183.

Lafferty, J., McCallum, A., & Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning, 8(June), 282–289.

Li, W., & Yang, M. Y. (2016). Efficient Semantic Segmentation of Man-Made Scenes Using Fully-Connected Conditional Random Field. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLI-B3(July), 633–640.

Lin, Y., Nex, F., & Yang, M. Y. (2018). Semantic Building Façade Segmentation from Airborne Oblique Images. International Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, ISPRS Technical Commission II Symposium 2018.

Liu, H., Zhang, J., Zhu, J., & Hoi, S. C. H. (2017). Deepfacade: A deep learning approach to facade parsing. IJCAI International Joint Conference on Artificial Intelligence, 2301–2307.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 3431–3440.

Martinović, A., Mathias, M., Weissenberg, J., & Van, L. (2012). A Three-Layered Approach to Facade Parsing. ECCV.

Rahmani, K., Huang, H., & Mayer, H. (2017). FACADE SEGMENTATION with A STRUCTURED RANDOM FOREST. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 4(1W1), 175–181.

Rahmani, K., & Mayer, H. (2018). High Quality Facade Segmentation Based On Structured Random Forest, Region Proposal Network And Rectangular Fitting. In ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences (Vol. 4, pp. 223–230).

Seif, G. (2018). Semantic-Segmentation-Suite. Retrieved from https://github.com/GeorgeSeif/Semantic-Segmentation-Suite

Teboul, O., Simon, L., Koutsourakis, P., & Paragios, N. (2010). Segmentation of building facades using procedural shape priors. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 3105–3112.

Tu, J., Sui, H., Feng, W., Sun, K., Xu, C., & Han, Q. (2017). Detecting building façade damage from oblique aerial images using local symmetry feature and the gini index. Remote Sensing Letters, 8(7), 676–685.

Xiao, J., Gerke, M., & Vosselman, G. (2012). Building extraction from oblique airborne imagery based on robust façade detection. ISPRS Journal of Photogrammetry and Remote Sensing, 68, 56–68.

Yang, M. Y., Förstner, W., & Chai, D. (2012). Feature evaluation for building facade images-an empirical study. In International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences: [XXII ISPRS Congress, Technical Commission I] 39 (2012), Nr. B3 (Vol. 39, No. B3, pp. 513-518).

Yang, M. Y., & Förstner, W. (2011). Regionwise classification of building facade images. In Conference on Photogrammetric Image Analysis (pp. 209-220). Springer.

Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? Proceedings of Neural Information Processing Systems - Volume 2 Pages 3320-3328.

Zhu, X. X., Tuia, D., Mou, L., Xia, G. S., Zhang, L., Xu, F., & Fraundorfer, F. (2017). Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. IEEE Geoscience and Remote Sensing Magazine, 5(4), 8–36.