International Cartographic Association
Association Cartographique Internationale

Application of Deep Learning for 3D building generalization

Yue Wu*, Yevgeniya Filippovska, Valentina Schmidt, Martin Kada

Institute of Geodesy and Geoinformation Science, Technische Universität Berlin,
yue.wu.1@campus.tu-berlin.de, yevgeniya.filippovska@tu-berlin.de, valentina.schmidt@tu-berlin.de, martin.kada@tu-berlin.de

* Corresponding author

Abstract: The generalization of 3D buildings is a challenging task, which needs to consider geometry information, semantic content and topology relations of 3D buildings. Although many algorithms with detailed and reasonable designs have been developed for the 3D building generalization, there are still cases that could be further studied. As a fast-growing technique, Deep Learning has shown its ability to build complex concepts out of simpler concepts in many fields. Therefore, in this paper, Deep Learning is used to solve the regression (generalization of individual 3D building) and classification problems (classification of roof type) simultaneously. Firstly, the test dataset is generated and labelled with the generalization results as well as the classification of roof types. Buildings with saddleback, half-hip, and hip roof are selected as the experimental objects since their generalization results can be uniformly represented by a common vector which aims to meet the compatible representation of Deep Learning. Then, the pre-trained ResNet50 is used as the baseline network. The optimal model capacity is searched within an extensive ablation study in the framework of the building generalization problem. After that, a multi-task network is built by adding a branch of classification to the above network, which is in parallel with the generalization branch. In the process of training, the imbalance problems of tasks and classes are solved by adjusting their donations to the total loss function. It is found that less error rate is obtained after adding a classification branch. For the final results, two improved metrics are used to evaluate the generalization performance. The accuracy of generalization reached over 95% for horizontal information and 85% for height, while the accuracy of classification reached 100% on the test dataset.

Keywords: generalization, 3D buildings, Deep Learning, convolutional neural networks (CNN)

1. Introduction

Digital 3D city models serve nowadays a wide range of application fields, such as urban planning, environmental simulations, navigation, location-based services, virtual 3D globes and 3D landscape visualizations, etc. (Biljecki et al., 2015). An essential component of such models are 3D buildings since they significantly influence the visual perception of the entire model. To efficiently manage the massive amount of 3D building data of an entire city, one option is to perform a cartographic generalization of these models to remove non-essential geometric details while preserving their original overall shape characteristics. In this context, the strict rectangular, parallel, and coplanar arrangement of the faces of the 3D building models (often given as polyhedra in boundary representation) as well as building part and roof symmetries, and object defining forms and components (e.g. church towers) are frequently mentioned. In addition to the management task itself, low detailed 3D building models reduce storage requirements, shorten geometric computations, and accelerate real-time visualizations (Kada, 2006).

With the widespread availability of area-wide 3D city models, various cartographic generalization methods for 3D buildings models have been proposed. This includes a multitude of approaches for simplification and, although less common, also for aggregation, symbolization and typification. Many of these generalization operators, however, involve sophisticated rules that are imprecisely defined and have been often developed specifically with respect to the geometric properties of the respective data sets to which they have been applied. They are rarely tested or quantitatively evaluated for large areas. It is not uncommon that unrealistic assumptions were made about the objects' shapes, so that these methods cannot be effectively adopted for other models. It is therefore not surprising that cartographic generalization of 3D building models has not progressed from academic studies to real-world applications yet.

In recent years, methods of Machine Learning and Deep Learning, the latter particularly in the form of convolutional neural networks (CNNs), have made enormous progress ever since Krizhevsky et al. (2012) presented AlexNet for image classification. Since then, neural networks have become an indispensable tool not only for computer vision, speech recognition, and natural language processing, but also for a wide range of related tasks and applications. It is therefore not surprising that these techniques have once again slowly made their way into cartographic generalization. One decisive obstacle, however, is the availability of suitable training data that is required in large quantities in order to apply these technologies effectively. As of today, training data, or the lack thereof, is a serious obstacle to the cartographic (3D) generalization of building models.

In this paper, we aim for a cartographic simplification of 3D building models by means of a symbolization task,

comparable to the concept as described by Thiemann & Sester (2006). A complex 3D building model is replaced by a 3D building template that is geometrically adapted to the original shape, in our case using a CNN. The biggest challenges of such an approach are the generation of a sufficient amount of training data, the definition of a 3D building template, and the CNN architecture specific to this task. Following a brief overview on related works on the task of cartographic 3D building model generalization and on Deep Learning in section 2, the preparation of training data and the 3D building template is described in section 3. The architecture of the CNN for classification and parameter regression of 3D building templates is then presented in section 4. In section 5, the training of the CNN network is described. Section 6 gives an analysis of the results before a conclusion is given and future work outlined in section 6.

2. Related work

The proposed generalization approach for 3D building models is closely related to cartographic simplification, for which related work is presented below. Furthermore, recent developments in the field of Deep Learning that are relevant are discussed both in the computer vision and cartographic (generalization) domain.

2.1 Cartographic 3D simplification

Over the past twenty years, many algorithms for 3D building model generalization have been proposed. The existing methods can be classified into several categories: (1) Using geometrical information for generalization: Thiemann and Sester (2006) used a generic parameterized template of a typical (residential) building and replaced the original 3D building models with symbolic versions that were instantiated with parameter values that best resemble the original shapes. In this way, a shape simplification in conjunction with emphasizing on the building object characteristics can be accomplished. Kada (2007) generalized building models by gluing building fragments that are the result of a space decomposition process along approximating planar half-spaces and used primitive instances of roof shapes to preserve their correct shapes. (2) Using mathematical morphology for generalization: Mayer (2005) and Forberg (2007) developed scale-space techniques for the automatic generalization of orthogonally shaped 3D buildings, partially based on the morphological operators opening and closing for 3D vector data. By comparison, Zhao et al. (2012) presented a mathematical morphology-based approach that can generalize complex 3D building models in a unified manner by using the semantic relationships between components. (3) Generalization based on CityGML: Baig et al. (2013) proposed a unified generalization framework to derive lower levels of detail (LoD) from higher LoD by taking semantics as well as geometric aspects of CityGML (Groeger, 2007) buildings into account. Fan and Meng (2012) presented a three-step approach to derive LoD2 buildings from CityGML LoD3 models by treating different semantic components of a building separately. The steps include simplifying wall elements, generalizing roof structures, and reconstructing the 3D building.

2.2 Deep Learning

Deep learning method is a popular research direction nowadays, which has brought revolutionary advances in a wide range of fields, such as image recognition, natural language processing, medicine, etc. In various deep learning methods, Convolutional Neural Networks (ConvNets or CNNs) is one of the main neural networks trends to solve image-related problems. So far, a multitude of different CNN architectures have already been proposed, e.g. AlexNet (Krizhevsky et al., 2012), VGG (Simonyan & Zisserman, 2014), GoogleNet (Szegedy et al., 2014), ResNet (He & Sun, 2014), which achieve more and more impressive performances for image classification and object detection, even exceeding human performance in recent studies. Additionally, CNNs are also designed for different types of problems. For example, SegNet is proposed for semantic pixel-wise segmentation, which can label each pixel with the class of its enclosing object region (Badrinarayanan et al., 2017). Mask RCNN is a deep neural network aiming to solve the instance segmentation problem, simultaneously giving bounding boxes, classes and masks for each object in an image (He et al., 2017).

2.3 Deep Learning for cartographic generalization

In the field of cartographic generalization, Machine and Deep Learning has been successfully applied. Lee et al. (2017) used different Machine Learning methods to classify buildings as a prior step for 2D building generalization. Sester et al. (2018) employed semantic pixel-wise segmentation solving the sub-problem of generalization on 2D building images. The prediction from their networks gives regular shapes which are then amendable for vectorization. Kudinov (2018) used Mask-RCNN to predict the polygon types and masks of roof segments, then 3D buildings were extruded by ArcGIS. Although the accuracy cannot reach the level of manual editing, it significantly reduced the manual labour cost by fine tuning the predictions instead of manually digitizing roof segments.

Concluding, it can be seen that cartographic generalization of 3D building models has long been a scientific topic, and Deep Learning method has achieved great development and been widely applied into different domains. However, using Deep Learning methods for the 3D building generalization problem still lacks focus, and to the authors' knowledge, there exist no research that combines Deep Learning techniques with 3D building generalization. Therefore, this paper proposes on a CNN method to generalize 3D buildings.

3. Data preparation

As mentioned earlier, CNN training requires a fairly large amount of labelled data. To the best of our knowledge, no such data is yet available for the task of 3D building

generalization. This problem is exacerbated by the fact that there are no algorithms that can satisfactorily solve the task of 3D generalization. We therefore limit ourselves to a small subset of building shapes for which we define a simple generalization based on existing 3D models in order to train our networks. The goal is to show the feasibility for this simple approach, and then later implement a generalization for more complex models, e.g. by extending it in a recurrent architecture.

As a data basis for the cartographic generalization and the preparation of training data, the 3D city model of Stuttgart, Germany, was used. It was semi-automatically reconstructed using a regular elevation grid generated by manual stereo analysis of aerial images as well as measured break lines (slopes, walls, ramps) and elevation points (peaks, valleys) (Wolf, 1999). The footprints were taken from the automated real estate map (ALK) that were intersected with a digital terrain model in order to obtain the elevation of the footprint coordinates. For the reconstruction of the roof structure, basic shapes such as flat, gable, hipped, pyramid, barrel, shed, domed roofs, etc. are selected from a library and the corresponding roof parameters are measured in the aerial images. The roof shape of complex buildings such as castles, towers or churches are assembled from these basic shapes. Due to the specific reconstruction method used, this type of 3D building models has very detailed floor plans and roof structures, but only flat façade elements without windows, doors, balconies, etc. The accuracy of the measured points is specified in (Wolf, 1999) as approx. ± 8-10 cm in position and approx. ± 15-20 cm in height.

Each 3D building model is first automatically generalized into the structure consisting of a rectangular footprint and a roof with a single ridgeline. This structure serves as the generalization results from which the neural network is supposed to learn its generalization process (as well as the classification of building type). Because CNN architectures are mainly built for image data, the models need to be converted and labelled for Deep Learning. Besides the preparation of the image and label, dataset selection is also important for learning.

3.1 Selection of building data

The buildings with saddleback, half hip, and hip roof (Figure 1) are selected as test datasets because the generalized structures of these three-type buildings can be uniformly represented (Referring to the preparation of the input labels).

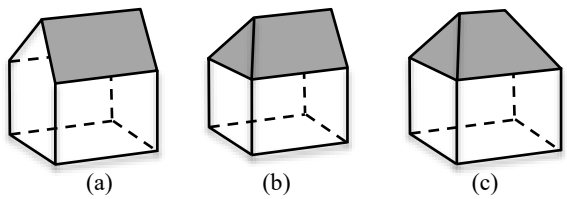


Figure 1. Three common roof types of buildings. (a) Saddleback roof; (b) half hip roof; (c) hip roof.

Deep Learning depends heavily on data. To ensure the quality of the learning, the buildings with rectangular or rectangular-like footprints should be eliminated because these buildings have quite simple generalization (almost the same with the original footprints) and occupy large amount, which causes an imbalance in the building dataset. Therefore, only those buildings with relative complex footprints are considered in the learning process. Figure 2 displays the process of judging the footprint complexity. As shown in Figure 2(c), the minimum area bounding box (MABB) of the footprint is proportionally shrank (here the ratio is 90%); if the shrank MABB is totally covered by its original footprint, this building should be eliminated from the experimental dataset. Based on this principle, in the example of Figure 2(c), the building is selected.

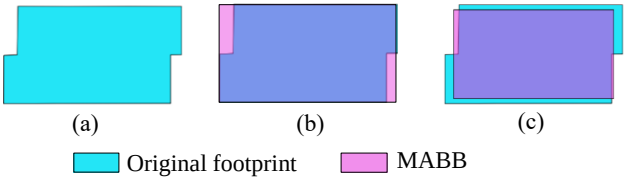


Figure 2. The process of judging building footprint complexity. (a) The original building footprint; (b) minimum area bounding box (MABB) of the footprint; (c) shrinking MABB.

3.2 Preparation of input rasters

To provide the data in a format compatible with CNNs, the boundary representation of a 3D building is converted into a raster. This conversion implies three processing steps, i.e. rotation, rasterization, and rendering.

(1) Rotating buildings into the canonical orientation (Figure 3(a)). Through rotation, the ridges of the buildings aligned to the horizontal axis. Beside preventing rasterization artefacts along the edges, another effect of the canonical orientation is that, no rotation invariants need to be learned during training.. Furthermore, the relation between X coordinate and Y coordinate can be enhanced, which can simplify the learning process and increase the training accuracy in some degree (Cohen et al., 2016).

(2) Rasterizing each building separately with grid cells values based on the corresponding height (Figure 3(b)). Each building is centered in the grid of 100*100 pixels. The resolution is kept the same in length and width. After adding 6 pixels padding around, the 112*112 grid is generated, which can be exactly scaled to the input of the network (224*224) without errors.

(3) Rendering the raster by colorbar. Colorbar, including a range of RGB colours, is used to map the height values in the raster to the corresponding colour (Figure 3(c)). The maximum and minimum height values correspond to the head and tail colours of the colorbar, respectively. Through this way, each grid can generate a colour image and the colour information in the image represent the relative height information. The

Proceedings of the International Cartographic Association, 2, 2019.
29th International Cartographic Conference (ICC 2019), 15–20 July 2019, Tokyo, Japan. This contribution underwent single-blind peer review based on submitted abstracts. https://doi.org/10.5194/ica-proc-2-147-2019 | © Authors 2019. CC BY 4.0 License.

correspondence relationship between colour and height is calculated by Equation (1):

$$color[i] = \frac{height - height.minimum}{height.maximum - height.minimum} \cdot n \quad (1)$$

where i is the ith colour, n is the total colour number in colorbar. The similar preprocessing work can be also seen in reference (Eitel et al., 2015).

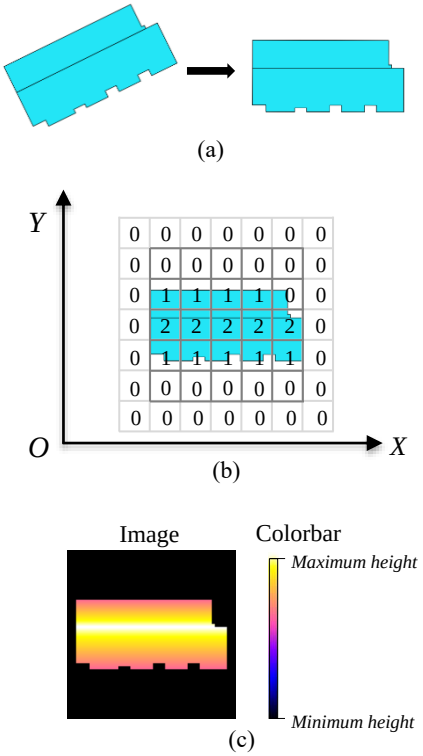


Figure 3. Preparation of input images. (a) Rotation; (b) rasterization example (Rasterizing one building into 5*5 grid with padding 1 pixel); (c) rendering building height by colorbar.

3.3 Preparation of input labels

To generate a training dataset for simultaneously solving for the two tasks in a supervised learning approach, two label vectors are defined for generalization and roof type classification, respectively.

(1) Input label of roof type classification

The target vector for roof type classification task is defined as follows:

$$Vector_{classification} = (roof_{type}) \quad (2)$$

The represented roof types in the dataset include saddleback roof, half hip roof, and hip roof and they are converted to one-hot encoding as input for our model.

(2) Input label of generalization

The target vector for building generalization task is described by the parametric shape of a building, which includes a building block and an eave (Figure 4). The parameters vector is obtained by calculating the minimum and maximum extend of the generalized footprint, the start and end points of the generalized ridge, and its eaves/ridge height ratio (Haala & Kada, 2010).

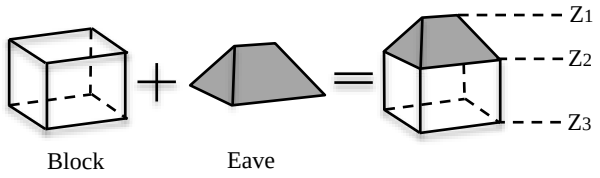


Figure 4. Parametric shape of a 3D building.

Firstly, a footprint is generalized into a rectangle described by four coordinates (X_{min_rec}, Y_{min_rec}) and (X_{max_rec}, Y_{max_rec}). Shown in Figure 5, the generalization process is described as followings: based on the raster image, the boundary pixels are detected and classified with respect to their position; then the coordinates of the generalized rectangle is calculated as average value of its corresponding boundary pixels. For example, X_{max_rec} is calculated based on the average value of its right boundary pixels. However, there are cases where few pixels have a displacement with respect to other pixels on the same direction (as shown in the circled area of Figure 5(b)), which can affect the generalization result; thus, they have been deleted based on a chosen threshold (0.33) as follows.

$$\frac{right.pixel.x - boundary.pixel.xmin}{boundary.pixel.xmax - boundary.pixel.xmin} < threshold \quad (3)$$

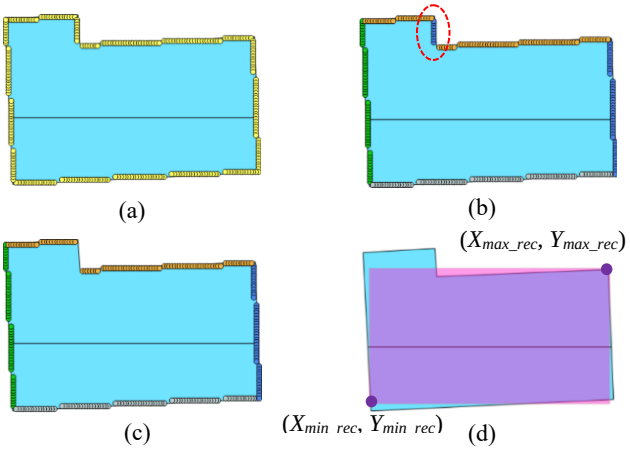


Figure 5. Generalization process of building footprint. (a) Extracting boundary pixels; (b) classifying boundary pixels by location; (c) deleting pixels according to thresholds; (d) generalized footprint with roof.

Secondly, the generalization of the roof shape preserves the main ridgeline and ignores all roof superstructures. Since the ridgeline is rotated into horizontal, its two terminal points can be represented by (X_{min_ridge}, Y_{ridge}) and (X_{max_ridge}, Y_{ridge}). Exceptionally, if the ridgeline starts or ends outside the footprint rectangle, the start or end point is moved inside the building, accordingly.

Proceedings of the International Cartographic Association, 2, 2019.
29th International Cartographic Conference (ICC 2019), 15–20 July 2019, Tokyo, Japan. This contribution underwent single-blind peer review based on submitted abstracts. https://doi.org/10.5194/ica-proc-2-147-2019 | © Authors 2019. CC BY 4.0 License.

Thirdly, the colours of the raster image represent the relative height, as described in section 3.2. Therefore, the parameter $roof_{ratio}$ is used to represent the eaves/ridge height ratio. Equation (4) is used to calculate $roof_{ratio}$:

$$roof_{ratio} = \frac{roof_{height}}{building_{height}} = \frac{Z_1 - Z_2}{Z_1 - Z_3} \quad (4)$$

where Z_1, Z_2, Z_3 are shown in Figure 4.

Based on the above three descriptions, the target vector for learning the generalization task is defined as:

$$Vector_{Generalization} = \begin{pmatrix} X_{\min\_rec} \\ X_{\max\_rec} \\ Y_{\min\_rec} \\ Y_{\max\_rec} \\ X_{\min\_ridge} \\ X_{\max\_ridge} \\ Y_{ridge} \\ roof_{ratio} \end{pmatrix} \quad (5)$$

The above two vectors, i.e. $Vector_{classification}$, $Vector_{generalization}$ are used as the input labels of the multi-task.

4. Networks architecture

We use ResNet50 pre-trained on ImageNet dataset as baseline network, since the pre-trained weights are beneficial for faster and better convergence. As the winner architecture in the LSVRC 2015, ResNet is derived from a simple deep convolution neural network by adding skip connections. It has been proved that this improved architecture allows for training very deep models by overcoming the problem of vanishing gradients, and thus benefits from the increase in accuracy provided by deeper models (He & Sun, 2014). The added shortcut connections form residual units in the network. In ResNet50, there are 16 residual units in total.

4.1 Ablation study

To examine how the network depth affects the performance, an ablation study with ResNet50 conducted. Thus, we training different shallower counterparts on the dataset, to search the network with optimal depth to the necessary feature representations. Here, the depth is represented by residual unit and counted from the input to the last layer (an activation layer, always following an add layer) of the residual unit, an example is marked with the pink colour in Figure 7. Since the generalization of 3D buildings is relatively more complex than the classification of roof type, we consider it as the main task and use it to train networks with various depths and monitor the corresponding training and validation loss values. As shown in Figure 6, the loss decreases with the increasing depth, approximately. The training loss is always higher than the validation loss, which indicates that there is no overfitting during the training process. The network with the '14th' residual unit before the last average pooling layer is considered as having the optimal capacity since deeper networks do not improve the performance.

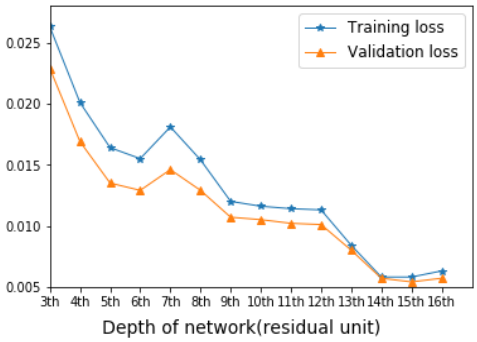


Figure 6. Training and validation losses of ResNet50 with different depths.

4.2 Multi-task architecture

The architecture of multi-task network is built by adding a classification branch, shown in Figure 7. It includes pre-trained layers of Resnet50 as feature extractor and is followed by a global average pooling layer (Lin et al., 2013) and two output layers (Sigmoid layer for regressing the parameters for building generalization; Softmax layer for roof type classification). The advantage of global average pooling layer lies in that it can achieves good performance both on regression as well as classification tasks (Zhou et al., 2016). The parameters of the feature extractor are initialized by parameters of ResNet50 trained on the ImageNet dataset. The parameters of the two output layers are randomly initialized.

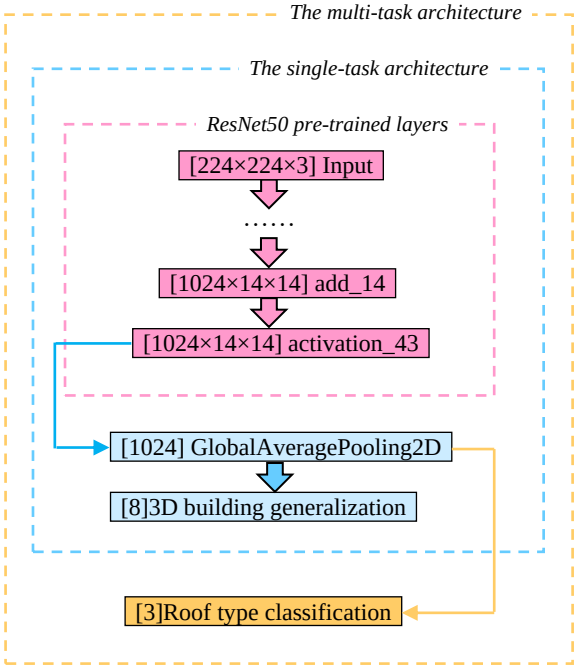


Figure 7. The multi-task architecture.

5. Training the network

5.1 Dataset description

We build a new training dataset using an existing data source of polyhedral 3D building models, which were photogrammetrically derived in a semi-automatic fashion, belonging to the area of Stuttgart, Germany. This area has 60138 buildings in total, including 22360 buildings with the saddleback, half-hip and hip roofs. By deleting the samples with rectangular or rectangular-like footprint shapes, 6358 samples are remained, including 4568 saddleback roofs, 912 half-hip roofs and 878 hip roofs. The percentage of the remained samples for training, validation and testing are 76.5%, 8.5% and 15%, respectively. All samples come with full annotation of the generalization result ($Vector_{generalization}$) and classification result ($Vector_{classification}$).

5.2 Tasks balancing

The multi-task training involves optimizing a global loss with two components: mean absolute error for the regression of the parameters for the 3D building generalization and categorical cross entropy for roof type classification task. However, to achieve a fast and good convergence, the two loss components must be balanced.

, Thus, the global loss function is the weighted linear sum of the two loss functions by Equation (6):

$$loss_{sum} = e \cdot loss_{generalization} + loss_{classification} \quad (6)$$

where the weights are set as e and 1 which performs the best during the training.

5.3 Class balancing

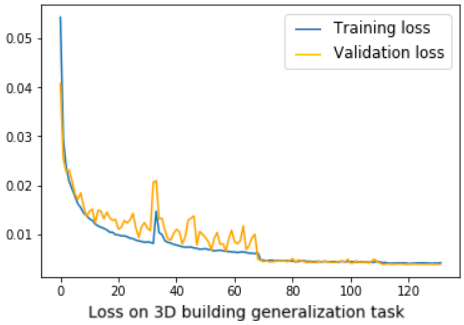
We face also a class imbalance problem for roof type classification, since the number of the buildings with saddleback roof is much larger than the other two represented classes. Therefore, the loss of classification is calculated as shown in Equation (7) by adding class-specific weight to each sample, aiming to balance the contributions of different classes to the classification loss.

$$loss_{classification} = \frac{\sum_i w_k \cdot loss_{i\_sample}}{n} \quad (7)$$

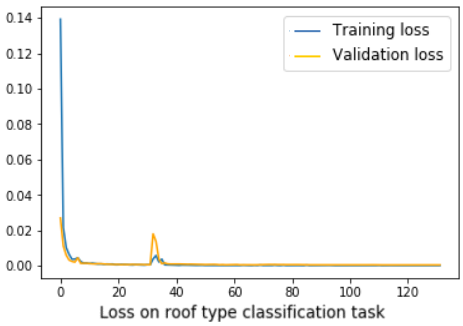
where $loss_{i\_sample}$ is the loss of the ith sample; n is the total number of samples in one batch; W_k=0.46, 2.68, 3.08, k=0,1,2, representing saddle back roof, half hip roof, hip roof, respectively. The weights are calculated based on the sample number in each class.

5.4 Experiment configuration

Stochastic Gradient Descent (SGD) is chosen as optimization algorithm of the global objective function and the parameters of SGD are set as: *momentum*=0.9, *nesterov=True* and a schedule decay of *1e-8*. The learning rate is initialized by 0.001. The learning rate would be reduced by a factor of 0.1 once there isn't improvement on validation loss data over 10 steps. As stopping condition is the lack of improvement of loss value on the validation dataset for 20 training iterations. Figure 8 shows the training and validation loss values for the two tasks. After adding a classification branch, the model performance for 3D building generalization task has been improved. The final loss value on the validation dataset improved from 0.0057 to 0.0039. Furthermore, the classification prediction accuracy of the multi-task network has reached 100% on the test data.



(a)



(b)

Figure 8. Training and validation losses for (a) 3D generalization task and (b) roof type classification task.

6. Analysis

Since we achieved perfect classification prediction on test data , the following analysis is focused on the 3D building generalization task. Two evaluation metrics are proposed for measuring the regression performance of our network, considering the existing metrics cannot be used directly.

Firstly, an improved confusion matrix is used to visualize the performance of the regression problem, which is inspired by the work on using classification systems in regression domains (Torgo, 1996). Confusion matrix is normally adopted to visualize the performance of a classification. After discretizing the continuous values by keeping the same number of decimals, the confusion matrix is available to visualize the difference between the true and predicted components. Specifically, the normalized coordinates are converted to integer pixels and the ratio is accurate to three decimals. After obtaining the confusion matrix, the values may not have the linear relation with the matrix rank number. Thus, new rows and columns are inserted to the corresponding position by

filling zero, whereby difference can be intuitively acquired from its location in the confusion matrix. Figure 9 shows the improved confusion matrix.

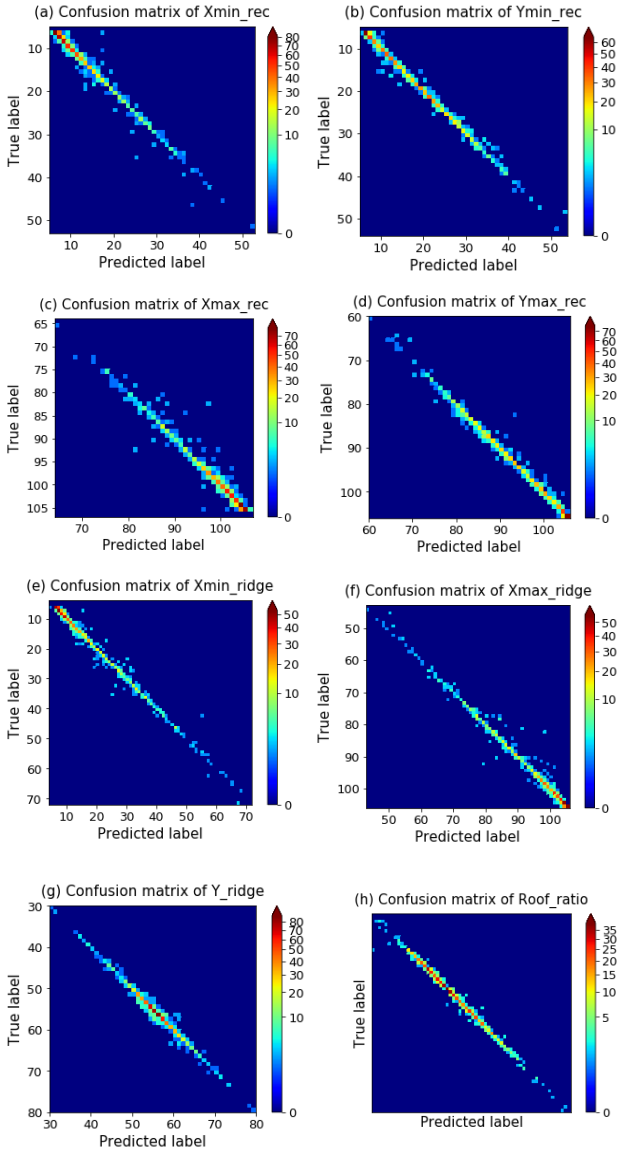


Figure 9. Improved confusion matrixes of building generalization vector. The colorbar represents the number of samples.

Secondly, the errors between the true and the predicted vertexes are measured by the percentage of detected vertices (PDV), which is similar to the percentage of detected joints (PDJ) used in human pose estimation, which judges correctly localized joints under normalized threshold (Toshev & Szegedy, 2014). However, in the rasterization process, buildings with different sizes are proportionally rescaled to the uniform scale; as a result, the error in the training process is uncorrelated with building size. Therefore, the localization precision is directly measured by the distance between the ground truth and predicted vertex locations without normalization by building size. Figure 10(a) lists the accuracy curves of four vertexes within different precision thresholds (pixel). For the ratio component, the threshold is set with the unit 0.001, a curve of the percentage of correctly predicted height ratio is shown in Figure 10(b) by varying thresholds.

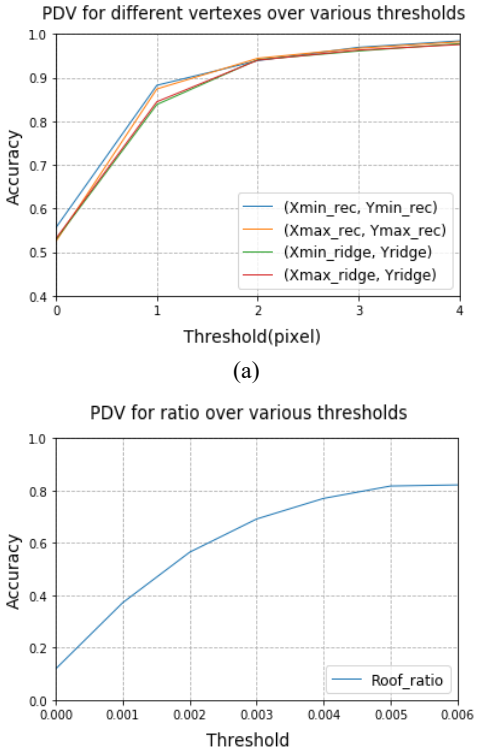


Figure 10. Accuracy of (a) four coordinates and (b) rood ratio over various thresholds.

7. Conclusion

Applying Deep Learning to solve the 3D building generalization as well as roof type classification simultaneously is a meaningful attempt, since Deep Learning has the ability to solve complex multivariate probability distributions. In the proposed method, the following five aspects should be noticed: (1) for regression problem, the completeness of dataset mainly reflects in containing enough data under different conditions instead of value distribution; (2) the capacity ablation of a pre-trained network is analysed to find necessary depth for feature representations; (3) multi-task learning can get better performance than single-task by balancing their losses; (4) the confusion matrix can be improved and used as metric for regression problem and can intuitionally reflect the difference between the true value and predicted value; (5) PDV is used as the performance metric when the value is uncorrelated with the size.

In the problem of generalization, the prediction result cannot be exact the same with the customized label, even with sufficient and various samples, just like cartographers cannot create totally the same generalization results. Nevertheless, the difference between deep learning prediction and customized label

may also provide some inspirations for the generalization evaluation.

Acknowledgements

The first author gratefully acknowledges the support of the Chinese Scholarship Council (CSC). This work was supported by the German Research Foundation (DFG) [grant number KA 4027/1-1].

References

Badrinarayanan, V., Kendall, A., Cipolla, R., and Member, S. (2015). SegNet : A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. arXiv:1511.00561v2 [cs.CV], 2015.

Baig, S.U., and Rahmann, A. (2013). A unified approach for 3D generalization of building models in CityGML. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL, 93–99.

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., Çöltekin, A.: Applications of 3D city models: state of the art review. In: ISPRS International Journal of Geo-Information, 4 (4), 2842-2889.

Cohen, J. P., Ding, W., Kuhlman, C., Chen, A., and Di, L. (2016). Rapid building detection using machine learning. Applied Intelligence, 45(2), 443–457.

Eitel, A., Spinello, L., and Riedmiller, M. (2015). Multimodal Deep Learning for Robust RGB-D Object Recognition. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany.

Fan, H., and Meng, L. (2012). A three-step approach of simplifying 3D buildings modeled by CityGML, International Journal of Geographical Information Science 26, 1091–1107.

Forberg, A. (2007). Generalization of 3D building data based on a scale-space approach. ISPRS Journal of Photogrammetry and Remote Sensing 62, 104–111.

Groeger, G., Kolbe, T.H., and Czerwinski A. (2007). Candidate OpenGIS City GML mplementation Specification. Open Geospatial Consotium Inc.

Haala, N., and Kada, M. (2010). An update on automatic 3D building reconstruction. ISPRS Journal of Photogrammetry and Remote Sensing, 65(6), 570–580.

He, K., Gkioxari, G., Dollar, P., and Girshick, R. (2017). Mask R-CNN. arXiv:1703.06870.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep Residual Learning for Image Recognition. In Proceedings of CVPR, 770–778.

Lin, M., Chen, Q. and Yan, S. (2013). Network in network. CoRR, abs/1312.4400.

Kada, M. (2006). 3D Building Generalization Based on Half-space modeling. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 2/w40), 58–64.

Kada, M. (2007). Scale-Dependent Simplification of 3D Building Models Based on Cell Decomposition and Primitive Instancing. Spatial Information Theory, 222–237.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Advances In Neural Information Processing Systems, 1–9.

Lee, J., Jang, H., Yang, J., and Yu, K. (2017). Machine Learning Classification of Buildings for Map Generalization. ISPRS International Journal of Geo-Information, 6, 309.

Mayer, H. (2005). Scale-spaces for generalization of 3D buildings. International Journal of Geographical Information Science, 19:8-9, 975-997.

Sester, M., Feng, Y., and Thiemann, F. (2018). Building Genearlization Using Deep Learning. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLII-4.

Simonyan, K., and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proc. International Conference on Learning Representations, 1–14.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., and Rabinovich, A. (2014). Going Deeper with Convolutions. CoRR, abs/1409.4842.

Thiemann, F., and Sester, M. (2006). 3D-Symbolization using Adaptive Templates. Proceedings of ISPRS Technical Commission II Symposium, (July), 49–54.

Torgo, L. (1995). Regression by Classification. in Proceedings of the 2nd International Workshop on Artificial Intelligence Techniques (AIT'95), eds. J. Zizka, and P. Brazdil, Brno, Czech Republic.

Toshev, A., and Szegedy, C. (2014). DeepPose: Human Pose Estimation via Deep Neural Networks. IEEE Conference on Computer Vision and Pattern Recognition, 1653–1660.

Wolf, M. (1999). Photogrammetric Data Capture and Calculation for 3D City Models. In: Fritsch, Spiller (eds.) Photogrammetric Week '99, Wichmann Verlag, Heidelberg, 305-312.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A. and Torralba, A. (2016). Learning Deep Features for Discriminative Localization. arXiv preprint arXiv:1512.04150.

Zhao, J., Zhu, Q., Du, Z., Feng, T., and Zhang, Y. (2012). Mathematical morphology-based generalization of complex 3D building models incorporating semantic relationships. ISPRS Journal of Photogrammetry and Remote Sensing, 68, 95–111.