A DEEP LEARNING STUDY OF EXTRACTING NAVIGATION AREA FROM CAD BLUEPRINTS

L. Niu[1, *], Y.Q Song[2], J. Su[1], H.M. Zhang[1]

[1] School of Surveying and Urban Spatial Information, Henan University of Urban Construction, 467036 Pingdingshan, China - (l.niu, 064415104, zhmz8)@hncj.edu.cn
[2] School of Geographic and Environmental Science, Normal University of Tianjin, West Bin Shui Avenue, 300387 Tianjin, China - syiq@hotmail.com

Commission IV, WG IV/4

KEY WORDS: Deep learning, Extraction, Navigation area, CAD blueprints

ABSTRACT:

Deep learning technology is a cutting edge topic of AI region, and draws more attention from photogrammetry and remote sensing society. In this study, we strive to combine deep learning with CAD designs to extract navigation area (room). To this, we mark more than 200 2D building blueprint in CAD forms to construct the learning set to train object detection model based on TensorFlow. This model is the faster R-CNN inception v2 model from COCO dataset. The test and result section is composed of three parts: First part demonstrates the model performance on learning dataset; second part applies the generated model to extract rooms from untrained raw CAD blueprints; Third part covers the comparison between deep learning extracted result and geometric based algorithm extracted result. Test result shows that the deep learning approach could achieve higher accuracy than geometric approach under regular shape situations. In conclusion, we have proposed a well-trained deep learning model that could be utilized to construct a schema of the navigation area for 2D CAD blueprints.

1. INTRODUCTION

1.1 Deep learning development

Deep learning technology is the most focused area among the information discipline. Its main purpose is to apply convolutional neural network (CNN) to the analysis of textual and image based cognition area. As for the remote sensing and photogrammetry community, image feature extraction is the most promising application scope. For the image feature extraction, CNN could be used to keep the balance between the main schema and fine details simultaneously for processing images. Besides, CNN has also shown advantages on crossing the barrier between standard extraction templates and complex real images, which is achieved by CNN tracking features on both high level and low level of image characteristics(Baglatzi and Kuhn 2013, Herrault, Sheeren et al. 2013, Richter, Richter et al. 2013, Zhang and Zhang 2017).

Despite of deep learning drawing high attention of our community, its limitation is easily perceived: Although we could apply parallel computing technology to increase its execution by the geometric growth, the computing burden introduced at the same time is with the exponential growth. For this, it is an extreme challenging task to achieve a solution balancing high accuracy, low cost and short time period.

Nevertheless, we could still receive an acceptable accuracy with limited computing resources by spending a comparative long time spread. This affordable solution is crucial for overcoming several bottlenecks that emerge by over-complexity problems especially existing in spatial feature related applications. A typical scene of this kind is the navigation area extraction from raw CAD blueprints.

There exist many efforts to apply deep learning methodologies to image processing fields like remote sensing to extract object with boundaries (Ball, Anderson et al. 2017, Tian, Zhang et al. 2017, Xu, Wu et al. 2018). These works strive to evaluate the extracted objects with shapes, topologies and other crucial feature both from spectral domains and spatial domains.

1.2 CAD navigation area extraction

CAD blueprints are widely used as an important source of building navigation data. Nevertheless, due to the complexity of the CAD blueprints, it is difficult to directly extract navigable area from them(Zhang, Mitra et al. 2010, Niu, Ke et al. 2012, Clementini and Ippoliti 2013, Herrault, Sheeren et al. 2013).

Existing solutions for this aim are composed of two main streams: One is introducing middle axis extraction; the other is applying occupation cell based solution. First solution requires the CAD blueprint to be thoroughly trimmed by removing guide lines and auxiliary lines, which are broadly used in the blueprints, and demands operators to delete critical connection objects from design maps, such as doors and stairways. Thus, the output of this solution is a basic skeleton of the navigation network.

Second solution is established by checking the occupancy for basic cells across the whole building. Say, if one cell is occupied by free space, it is marked as navigable; if it is occupied by obstacles like furniture, it is marked as non-navigable. The product by this solution is always a matrix of cell attribute. Therefore, the spatial resolution of cell significantly affects the data storage volume. Following this, if a fine-detail navigation map is demanded, a huge data volume is expected.

---

* Corresponding author

We could perceive that current navigation data extraction methods for CAD blueprints are with prominent flaws:

1. Tedious artificial interference for data preparation is mandatory

CAD blueprints have to be carefully processed in order to meet the demand of extraction solutions.

2. Prone to data errors

If any error in data occurs, the quality of generated navigation area could be significantly affected.

To tackle these problems, we strive to introduce deep learning technology like CNN to help stakeholders extract navigation area from 2D CAD blueprints. Our approach is expected to generate navigable area from normal 2D CAD blueprints without any additional edit and with error tolerance.

1.3 Paper organization

Our manuscript is organized as follows: Second part describes the sole principle of our methodology. Third part demonstrates the model performance on learning dataset, the experiment result on images from web source and our source. Furthermore, the comparison between our solution and existing solution is also provided. The forth part discusses the experiment output and evaluates quality of our solution. The final part concludes the whole paper.

2. CAD NAVIGTION AREA EXTRACTION BY DEEP LEARNING

2.1 CAD Navigation Area Extraction

2.1.1 Principle

Like normal deep learning applications, introducing this model to extract the navigable area of CAD blueprints have to fully utilize underlying relationships that are composed of the propagation and back-propagation between different neural network layers (figure1).
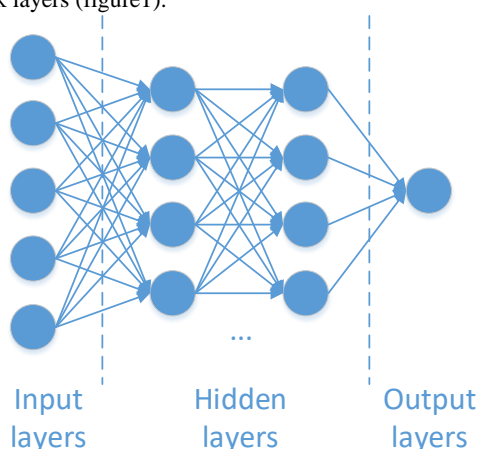

Figure 1. Illustration of the processing model for deep learning

Nevertheless, following this approach, the general neural network suffers from low computing performance against images containing complex objects. Thus, many hypothesis for processing images are proposed. One of them is the region proposal: it hypothesizes that many rectangular boxes in the processing scenes exist, to reduce the computing burden(Ren, He et al. 2015) (figure 2). This approach could be introduced to the object detection applications, for most of objects could be wrapped by rectangular boxes.

2.1.2 Bottleneck

As is widely known that existing CAD extraction solution lack of flexibility to properly process non-navigation related objects. Say, they cannot filter out guide lines and auxiliary lines, besides they require tedious human intervention to remove connectors, which prevents them from freely joining all the navigation space.

Although there requirements could be fulfilled, noticeable amount of human effort have to be introduced. This could draws significant economic burden, and cause the CAD processing task occupy large portion of the whole project schedule.
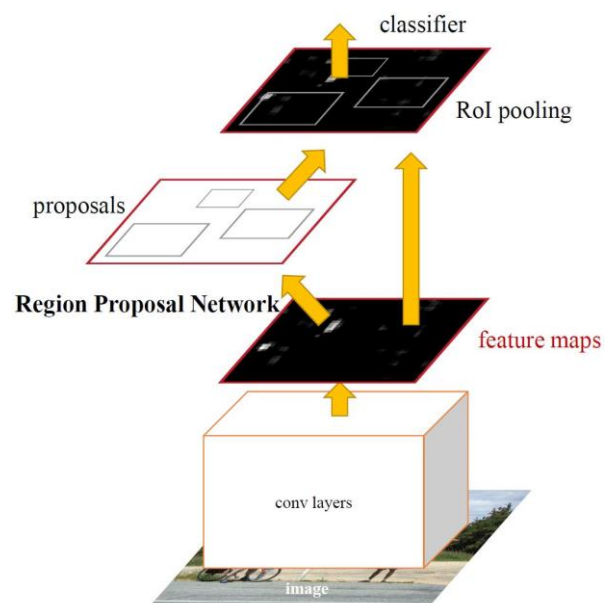

Figure 2. Illustration of the faster R-CNN model (by courtesy of Cornell University Library)

2.2 Deep Learning Approach

Deep learning technology is drawing more attention than ever, and its impact on image processing related field is prominent. Due to the CAD blueprints could be transformed into digital images, it is also possible to apply this method on CAD navigation space extraction.

For deep learning is mimicking the learning process of human, it is natural to count on its intellectual ability to tackle the complex situation pervasive in current blueprints. Therefore, the pre-processing of CAD blueprints targeting for fine details are unnecessary.

Due to that the core mechanism of deep learning is well explained by many famous works and tutorials, we only provide a problem-focused introduction aiming for navigation space extraction of CAD blueprints.

2.2.1 TensorFlow Solutions

TensorFlow is a well-known deep learning API provided by Google Inc.(tensorflow.org 2018). This solution is of high flexibility and compatibility. In our approach, it is introduced to establish a core schema with a time limit of one month, and this time period is suitable for testifying the feasibility of deep learning principle for CAD navigation space extractions.

The core idea of TensorFlow API is to establish a graph workflow processing numeric computations. In this graph, nodes represents mathematic operations, and edges connecting them demonstrate the data matrix, which is described by tensors, between them. It consists of many sub-packages and sub-APIs. The most related APIs within TensorFlow are slim API and object-detection API. The former one aims for general image processing purpose, and the latter one focuses on the object extraction from images.

Several typical TensorFlow models are demonstrated in table 1, which are provided by the TensorFlow detection model zoo data*. We could perceive that usually the increase of calculation time could improve the object detection performance. Nevertheless, when a threshold is passed, say 50ms here, selected models' performance are not significantly increased. When the increase of time spending crosses 100%, the increase of passing test quantity only exceeds 25%. Therefore, a balance between the accuracy and time cost should be carefully studied.

Model name	Speed (ms)	COCO mAP**	Outputs
ssd_mobilenet_v1_coco	30	21	Boxes
ssd_inception_v2_coco	42	24	Boxes
faster_rcnn_inception_v2_coco	58	28	Boxes
faster_rcnn_resnet50_coco	89	30	Boxes

Table 1. Metadata of selected object detection model in the TensorFlow model zoo

2.2.2 Implementation of CAD Navigation Area Extraction

As for the ease of our implementation of the deep learning solution, it is better to tune a TensorFlow model instead of constructing a novel one. Because the core mechanism of deep learning solution is to provide the model with proper learning materials, and let it progress gradually. Namely, this terminology advocates us to supervise an existing TensorFlow model to advance step by step, just like human learning progress.

To this, it is mandatory to select an appropriate set of candidate pictures to train the TensorFlow model. Thus, we plan to select free distributed pictures across main web searching engine. Besides, both considering time efficiency and extraction efficiency, the chosen model should also be fit for learning and processing images in a reasonable time period for our purpose, whose time performance is no worse than an average human

* Tensorflow detection model zoo, detail information by url: https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md

** MSCOCO Evaluation Protocol, detail information by url: http://cocodataset.org/#detections-eval

operator, according to our study. Say, the learning process should be shorter than one day, and the extraction time cost should also be measured by minutes.

Finally, by evaluating all mainstream models in the TensorFlow object detection model zoo, the only promising candidate model for CAD navigation space extraction are faster_rcnn_inception_v2_coco and faster_rcnn_resnet50_coco. Nevertheless, the latter one runs out of graphic card memories during test. Therefore, we only select the faster_rcnn_inception_v2_coco as current model.

This model first proposes a hypothesis of many existing regions in the considering image. And then it utilizes the convolutional layers and pooling layers pervasive in the R-CNN approach to testify whether its hypothesis is true. In detail, this hypothesis is always compared with the human marked images, and finally produce a neural network that could receive high scores generated by comparing its result to ground truth. This process is usually achieved by giving bonus when the extraction result is approaching ground truth and receiving penalties when the result is contradicting ground truth.

3. EXPERIMENT AND DATA

3.1 Experiment introduction

The whole test workflow is straightforward: we train the selected faster_rcnn_inception_v2_coco model by the CAD blueprints from internet, and then evaluate its extraction result qualities.

Aiming for testifying the generated model, we carefully filter out 19 candidates from the collected 2D CAD blueprint images from Baidu searching result. This is for avoiding duplicate images with learning images brought by Google searching engine. At last, all images in learning set are typical in either white/black background with clear building outlines. Due to the heavy task burden and our intention to evaluate solution robustness, all support lines and auxiliary lines in original pictures are kept. Besides, there is another test image source generated by transforming existing CAD vector blueprints to raster data format. These images are introduced to compare the generated navigation extraction result with the existing geometric algorithm based solutions.

3.2 Test Image Result

3.2.1 Deep learning model training data

As shown from figure 3 to 8, all collected data about the faster R-CNN model are provided. Figure 3 shows the classification loss of the box classifier; figure 4 provides the localization loss of the box classifier; figure 5 gives the object-identification loss of the region proposal network; figure 6 illustrates the localization loss of the region proposal network; figure 7 demonstrates the time cost distribution for the global step; figure 8 provides the total loss of the utilized model.

3.2.2 Extraction Result on All Test Images

Figure 9 illustrates all the generated 20 images in a large canvas. We could perceive that extracted images cover noticeable large area of all test area. All these extracted regions are marked with high confidence (usually larger than 90%), and only once below 90% in e. Nevertheless, the deep learning solution fail to extract

several navigation area for most area with complex auxiliary lines and furniture.

3.2.3 Comparison between Deep Learning Solution and Geometric Solution

The comparison between the deep learning solution and geometric solution is shown in figure11 and table 2(Niu and Song 2016). As can be easily acknowledged: the geometric solution just extracts every rectangular sub-space in the experiment CAD blueprint, and the deep learning solution just provides navigable area where walls perfectly wrap the navigable regions. In other words, the former one provide more navigable space and the later one provide less navigable space compared to the ground truth. We will have in-depth discussion of the generated result in the next section.
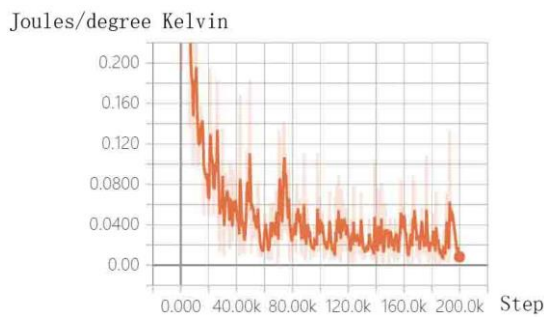

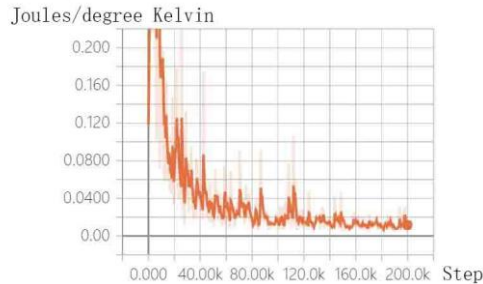Figure 3. Box classifier classification loss of the extraction result

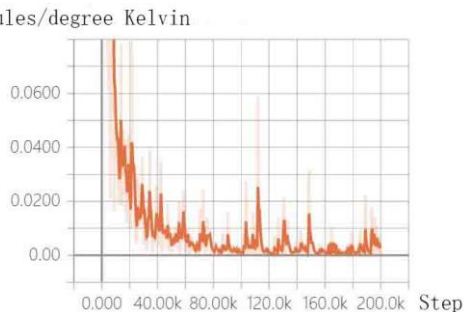Figure 4. Box classifier localization loss of the extraction result

Figure 5. Region proposal network objectness loss of the extraction result
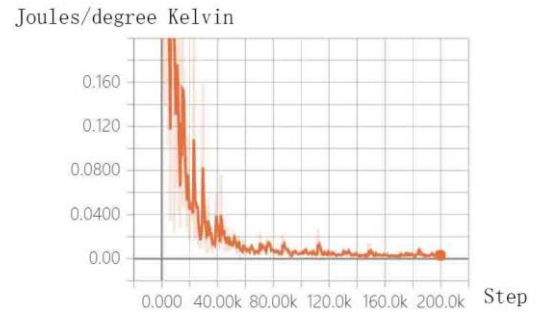

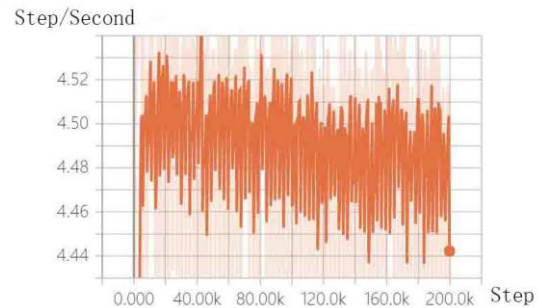Figure 6. Region proposal network localization loss of the extraction result
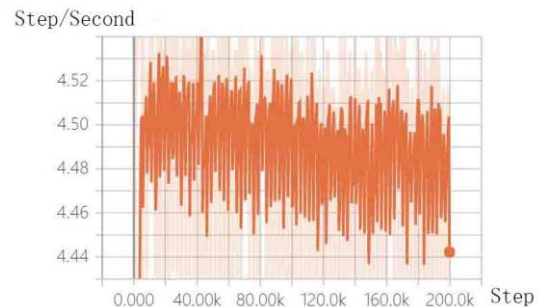

Figure 7. Global step time cost

Figure 8. Total loss of the extraction result

Method	Extracted Rooms	Correctly Extracted Rooms*	Correctly Extracted Ratio	Correct Ratio
Deep learning approach	22	20	90.9%	58.8%
Geometric approach	64	11	32.4%	32.4%

Table 2. Extracted rooms result comparison between the deep learning approach and geometric approach

4. RESULT DISCUSSION

The focus of this paper is not on the improvement of the deep learning model, so we provide only a general analysis of the learning data result. Due to the total step setting of the faster-RCNN is 20,000, all losses converge to 0 before this setting. Nevertheless, the classification loss of box and the objectness loss of RPN have higher frequency of oscillation and larger amplitude of vibration. Besides, the time cost of each global step is around 4.49 seconds.

---

* Several special types of room shape space are not considered currently, such as stairway space and corridor space.

As is demonstrated in figure 9, most of extracted navigation results generally describe the main regular navigable space in the provided CAD blueprints. Especially, when the wall boundaries are clearly marked, and the affiliation subjects like furniture and auxiliary lines are clearly distinguished with walls, the extracted results are good (d, f, g, h, i, k, l, m, n, o, p, q, r, s). Nevertheless, when wall boundaries are linked, and sometimes fused with decorated materials and furniture, the extracted result is not quite good (a, b, c, e, j, t).

Thus, we could conclude that most results have demonstrated the deep learning solution is capable of extracting navigation space with wall lines surrounding the building space. Besides the feature that deep learning solution could easily extract comparative small rooms, the deep learning solution could also be trained to identify the inner space and the outdoor space of buildings, which is also perfectly wrapped by walls. This studying process has also been observed which is just like the learning process of the by human intelligence.

Other than the positive performance of deep learning solution, the improper handling of the navigable space containing complex objects like furniture has shown its limitations (figure 10 and figure 11). Deep learning model could not identify whether an open space surrounded by several closing navigable area is passable. This problem could exaggerate when this type of space is also the neighbour of outdoor space. Moreover, this model also suffers from navigable space formed by small segments that is pervasive in stairway room and furniture. But we can explain the poor performance of this model by disclosing another truth: experienced human operators could confirm the existence of navigable space among these challenging situation for deep learning models, however, they also struggle to identify the exact geometric boundaries on this type of region.

When the experiment comes to the comparison between geometric algorithm and deep learning model, we could easily find that the geometric algorithm strictly follows the pre-defined rules for extracting navigable space, and certainly generates low-level mistakes by marking the open impassable space between walls. While the deep learning model only produce the area with high confidence, but leaves many real navigable space unmarked. Although both methods miss several small rooms, human operators may also not be able to extract these rooms without glancing the boundaries for quite a long time and lacking of confidence on the extracted results.
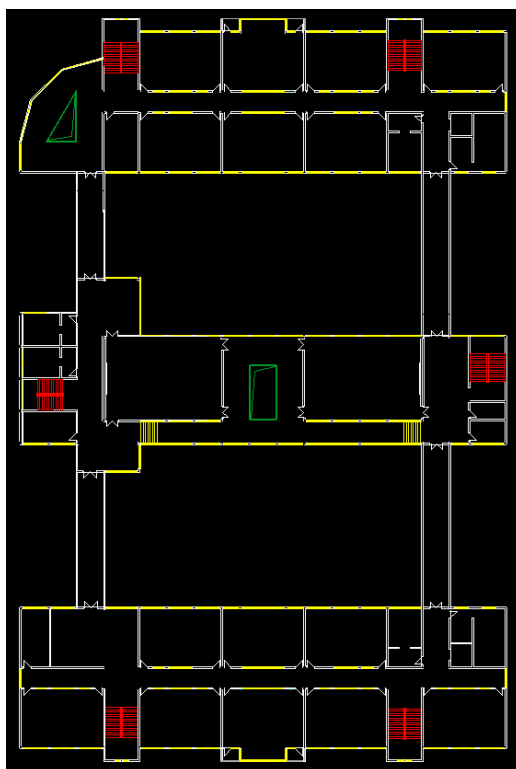
Figure 9. Illustration of navigation extraction result for CAD blueprints

Figure 10. Demonstration of the comparing CAD blueprint
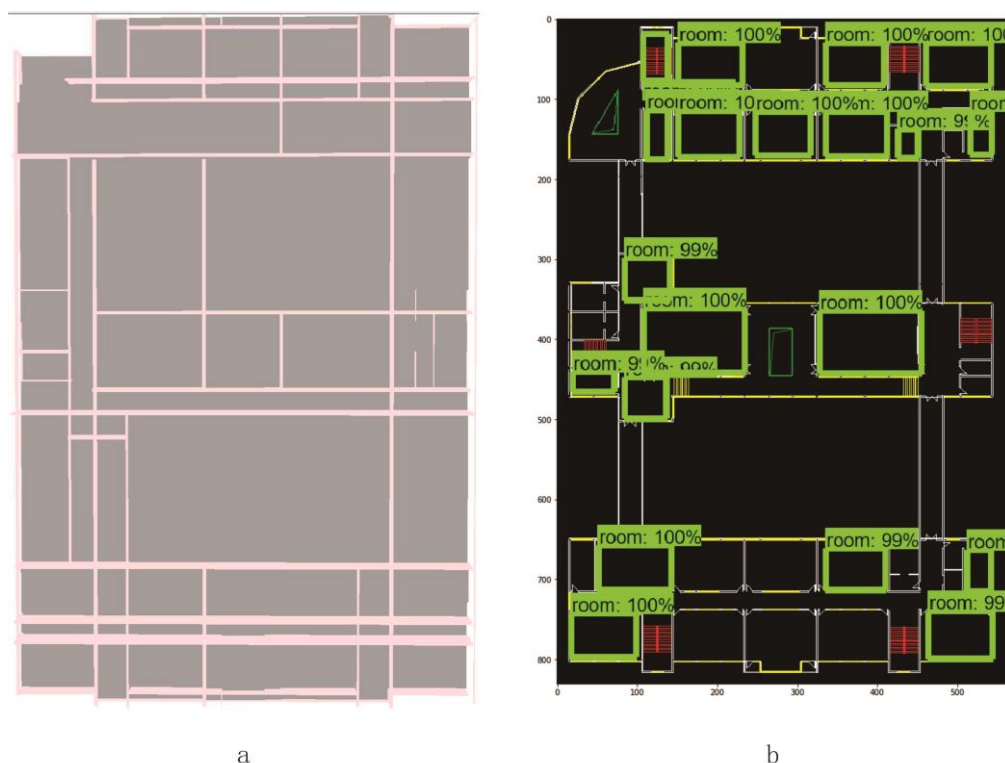

a   b

Figure 11. Illustration of navigation extraction result between geometric method and deep learning method for CAD blueprints

5. CONCLUSION

As is generally understood, deep learning models are flexible for handling a wide scope of complex image extraction scenes. Thus, its application for CAD navigation space extraction is reasonable. Our effort to apply existing TensorFlow models has illustrated that this approach has potential to tackle CAD blueprints with irregular spatial distribution of complex shape objects. We could perceive that the introduced deep learning model could identify quite complex scenes consisting of furniture and auxiliary lines.

Nevertheless, the deep learning model is also with noticeable flaws:

1. Its performance is unstable. Several regions containing furniture and stairways are extracted, but several navigable regions containing free space with vague semantic boundaries (these boundaries belong to neighbouring closing area and not forming regular walls) are ignored.

2. The learning data requirement is difficult to achieve. We have collected all 2D CAD blueprints through Google searching engine seems not to cover special cases implemented with local standard, especially non-standard objects in Chinese CAD blueprints.

For the next step of research, we have to collect enough amount of CAD blueprints implemented with both international and local standard to testify the flexibility of deep learning model.

ACKNOWLEDGEMENTS

REFERENCES

Baglatzi, A. and Kuhn, W., 2013. On the formulation of conceptual spaces for land cover classification systems. *Geographic information science at the heart of Europe*, Springer**:** 173-188.

Ball, J. E.; Anderson, D. T. and Chan, C. S., 2017. "Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community." *Journal of Applied Remote Sensing* **11**(4): 042609.

Clementini, E. and Ippoliti, E., 2013. Automatic extraction of complex objects from land cover maps. *Geographic Information Science at the Heart of Europe*, Springer**:** 75-93.

Herrault, P.-A.; Sheeren, D.; Fauvel, M. and Paegelow, M., 2013. Automatic extraction of forests from historical maps based on unsupervised classification in the CIELab color space. *Geographic Information Science at the Heart of Europe*, Springer**:** 95-112.

Niu, L., X. Ke and Z. Qiu, 2012. A pedestrian accessible position extraction method of existing 3D files for large building evacuations. *XXII ISPRS Congress*, Melbourne, Australia: 339-344.

Niu, L. and Song, Y., 2016. A schema for extraction of indoor pedestrian navigation grid network from floor plans. *23rd International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences Congress*, Prague, Czech, International Society for Photogrammetry and Remote Sensing: 325-330.

Ren, S.; He, K.; Girshick, R. and Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*: 91-99.

Richter, D.; Richter, K.-F. and Winter, S., 2013. The impact of classification approaches on the detection of hierarchies in place descriptions. *Geographic Information Science at the Heart of Europe*, Springer**:** 191-206.

tensorflow.org, 2018. "TensorFlow home page." Retrieved March 3rd, 2018, from https://www.tensorflow.org/.

Tian, S.; Zhang, Y.; Zhang, J. and Su, N., 2017. "A Novel Deep Embedding Network for Building Shape Recognition." *IEEE Geoscience and Remote Sensing Letters* **14**(11): 2127-2131.

Xu, Y., Wu, L.; Xie, Z. and Chen, Z.; 2018. "Building Extraction in Very High Resolution Remote Sensing Imagery Using Deep Learning and Guided Filters." *Remote Sensing* **10**(1): 144(1-18).

Zhang, L. and Zhang, L., 2017. "Deep Learning-Based Classification and Reconstruction of Residential Scenes From Large-Scale Point Clouds." *IEEE Transactions on Geoscience and Remote Sensing*: 1-11.

Zhang, X., Mitra, P.; Klippel, A. and MacEachren, A., 2010. Automatic extraction of destinations, origins and route parts from human generated route directions. *Geographic Information Science*. Heidelberg, Germany, Springer**:** 279-294.