Rotated Rectangles for Symbolized Building Footprint Extraction

Matt Dickenson Lionel Gueguen
UBER Technologies Inc.
mattd@uber.com, lgueguen@uber.com

Abstract

Building footprints (BFP) provide useful visual context for users of digital maps when navigating in space. This paper proposes a method for extracting and symbolizing building footprints from satellite imagery using a convolutional neural network (CNN). The CNN architecture outputs rotated rectangles, providing a symbolized approximation that works well for small buildings. Experiments are conducted on the four cities in the DeepGlobe Challenge dataset (Las Vegas, Paris, Shanghai, Khartoum). Our method performs best on suburbs consisting of individual houses. These experiments show that either large buildings or buildings without clear delineation produce weaker results in terms of precision and recall.

1. Introduction

Visual context helps users of digital maps to perform better at navigational tasks [13]. Building footprints (BFP) are an especially valuable source of contextual information, helping users to orient themselves in space [9]. In the case of ride-sharing applications, such context is helpful for drivers and riders to coordinate pick-up and drop-off locations in addition to navigation. However, the volume of buildings in even modestly-sized cities quickly grows intractable for manual map-making processes. Leveraging the long-tail distribution of popular locations by focusing on central business districts is not a viable solution, either: according to publicly released data from one popular ride-sharing platform, 40 percent of trips in Paris started in one suburban zone and ended in another [15]. Consequently, an automated method of mapping building footprints is valuable for ride-sharing applications. In this paper we describe an application of deep neural networks to the problem of detecting and symbolizing building footprints. The method described here allows for estimating rotated rectangles and merging them to create more complex polygons. This model is both more flexible than popular grid-aligned approaches [12] [11] and less computationally expensive than pixel-wise segmentation models [5] [2]. Section 2 discusses the current state-of-the-art for automated building footprint modeling. Section 3 then describes the architecture of our convolutional neural network (CNN) model for predicting rotated rectangles in more detail. Section 4 presents our experimental results. Section 5 concludes the paper by discussing potential extensions of this work.

Figure 1. Left. A training building polygon overlaid on satellite imagery. Middle. The best fitting rotated rectangle. Right. Parametrization of the best fitting rectangle with 5 degrees of freedom.

2. Related Work

Automated mapping of urban settings has been an important area of computer vision research for over two decades [4] [8] [7]. The present work builds on three related areas of research: pixel-wise semantic segmentation; object detection using grid-aligned bounding boxes; and text detection with arbitrary orientations. One recent development that has improved segmentation-based approaches is to classify pixels according to their distance from object boundaries (instead of predicting a segmentation mask), which helps to preserve boundary information [16] [5] [2]. Another approach is to predict bounding boxes of objects of interest. [12] uses manually chosen priors for bounding boxes so that the neural network itself only has to predict offsets (instead of coordinates). Building on [10], [11] relies on machine-learned anchor boxes and predicts both bounding boxes and object classifications. However, cities do not exhibit a perfectly grid-aligned pattern in their buildings, which is a major drawback of this approach for BFP extraction. Research on text detection in natural scenes helps to address this shortcoming, since text often appears at arbitrary orientations and sizes. [17] extend skew correction approaches to detect the orientation angle of text. Rather than treating orientation information as one step in a multi-stage pipeline,

[18] introduce a network that predicts the text geometry directly (either as a rotated rectangle or arbitrary quadrangle). By combining key features of these approaches—boundary-preserving distance transformations, learned attention, and rotated rectangle prediction from convolutional features—we propose that higher-quality BFP results can be automatically extracted from satellite imagery. The next section describes this approach in greater detail.

3. Proposed CNN Architecture

3.1. Rotated Rectangles

Building footprints are typically represented by polygons, where the number of points vary from 3 to a dozen in most cases. Given a polygon satisfying these assumptions, we simplify it to the best fitting rotated rectangle using the minimum-area encasing rectangle algorithm [3]. The algorithm proposes a discrete number of orientation; at each orientation, the bounding box rectangle of the original polygon is computed, and its area is recorded. Finally the rectangle achieving the smallest area is selected as the best fitting rotated rectangle. Such a simplification is illustrated in Fig. 1.

Each best-fitting rotated rectangle is described by 5 parameters: its center x, y, its width and height h, w and its angle with respect to a horizontal line α. As in region proposal networks and bounding box regression models, these parameters are further transformed with respect to some axis aligned grid cell characterized by a center x_g, y_g and dimensions h_g, w_g. The angle exists in the range $[-\pi/2, \pi/2]$, and it presents a discontinuity at the interval boundaries. This makes any regression against it harder to represent. Thus, we project the angle on the unit circle by using its two-dimensional cosine and sine representation. The best fitting rotated rectangle is then represented by the following 6 parameters given some grid g:

$$\hat{x} = x - x_g \quad ; \quad \hat{y} = y - y_g, \tag{1}$$

$$\hat{h} = \log h/h_g \quad ; \quad \hat{w} = \log w/w_g, \tag{2}$$

$$\hat{\alpha}_c = \cos 2\alpha \quad ; \quad \hat{\alpha}_s = \sin 2\alpha. \tag{3}$$

One of the final network layers is responsible for estimating these 6 parameters per grid cell provided, and a cell-based confidence metric indicates the presence of a building.

3.2. Grid Selection

In the context of object detection, recent region proposal based architectures [10], [11] grid the input raster image, and let each grid cell propose a candidate. Each grid cell thus becomes responsible for predicting the presence of an object, and the parameters of that object. In the context of building footprint extraction, we design the grid such that at most one building can be predicted by a cell. The grid is characterized as follows. For a VHR satellite image of resolution .5m and a minimal building size of 5×5 m^2, a cell shall be smaller than the minimum building size. Given typical convolutional neural network architectures, the grid dimension shall be proportional to the input dimension size by a factor that is a power of two. Given typical image input dimension of 512×512 at a resolution of .5m and a network that downscales these dimensions by a factor of $2^3 = 8$ to a grid of dimension of 64×64, the resulting grid cell size covers 4×4 m^2.[1]

3.3. Non-Maximal Suppression

As in most region proposal networks, a non-maximal suppression stage is required to remove the overlapping predicted rotated rectangle. The usual algorithm relying on quick computation of union and intersection areas needs special attention as rotated rectangles require more computation. We use an R-Tree spatial index [6] in order to avoid trivial computations when two polygons do not intersect. Given a rotated rectangle $r_i = (x_i, y_i, w_i, h_i, \alpha_i)$, it is first represented by a generic polygon made of four points. These points allow easy access to the axis-aligned bounding box (l_i, b_i, r_i, t_i) representing its left, bottom, right and top coordinates. Each polygon is associated to a score s_i produced by the network. The algorithm keeps the highest scored polygons, while removing high overlaps. Thanks to the R-Tree spatial index, each rotated rectangle is compared to a small subset of other polygons which potentially intersect it, greatly reducing the computational complexity.

3.4. Overall Architecture

Any fully convolutional neural network stack can be selected as precursor to the the overall model. In this paper, we select the first three blocks (six layers) from VGG-16 [14], which lead to a downscaling by 8 of the input image dimensions corresponding to the selected grid dimensions. Then that layer is fed in to two convolutional layers: one layer predicting the presence of a building in a cell, the second layer predicting the rotated rectangle parameters if present. The purpose of the second layer is that its gradients are only propagated on the cells which contain a building footprint (from the training data). We make use of mean squared loss to optimize both layers, with a balancing tradeoff that we set manually. The architecture is depicted in Fig. 2. During inference, the first layer allows us to identify the cells containing building footprints. Then, at the locations where objects are present the rotated rectangles are decoded from the estimated parameters provided by the second layer.

[1] The challenge data was presented at 650×650 resolution. We transformed the imagery to 512×512 for training and inference, and then re-projected the results into the original resolution for evaluation.

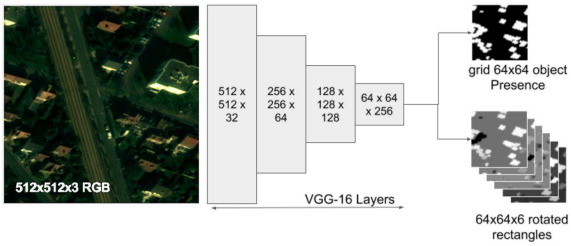

Figure 2. The overall CNN architecture proposed. The image is first passed through a standard CNN network, before being split into an object presence indicator and an object parameter estimator.

Table 1. Precision and Recall computed for an IOU above %50 on the training set for the four AOIs pertaining to the DeepGlobe challenge. F1 score is derived from precision and recall measurements as well.

@IOU 0.5	Precision	Recall	F1
Las Vegas	0.753	0.593	0.664
Paris	0.333	0.258	0.291
Khartoum	0.243	0.161	0.194
Shanghai	0.125	0.082	0.099
Total	0.364	0.273	0.312

4. Experiments and Results

4.1. DeepGlobe Challenge

We employed our approach in an experiment on the DeepGlobe building detection challenge [1], encompassing four AOIs covering the cities of Las Vegas in the U.S., Paris in France, Khartoum in Sudan and Shanghai in China. The training dataset, being organized by AOI, allows us to train city-specific weights by fine-tuning from a model trained on five U.S. cities. Given that the resulting footprints are associated with a confidence score, we determined a best threshold for each AOI to achieve maximum F1 score on a held-out dataset. The results obtained by the proposed approach are summarized in Tables 1 and 2, where Recall and Precision are computed for an Intersection Over Union (IOU) threshold of 50%. Clearly the F1 scores vary greatly across AOI, achieving the best performance in Las Vegas, and the worst performance in Shanghai. Illustrations of extracted and symbolized BFPs are provided in Fig.3, allowing us to discuss the performance metrics qualitatively. The proposed approach provides good approximations of small and well-separated buildings which are dominant in U.S. cities. However, when buildings are close to each other, of larger size or when they have non-rectangular shapes the proposed approach does not allow to capture them with a reasonable IOU, explaining lower F1 scores in Khartoum and Shanghai.

Table 2. Precision, Recall, and F1 score for each AOI in the validation set.

@IOU 0.5	Precision	Recall	F1
Las Vegas	0.760	0.601	0.671
Paris	0.323	0.257	0.286
Khartoum	0.253	0.167	0.201
Shanghai	0.132	0.084	0.103
Total	0.364	0.273	0.312

Figure 3. The rotated rectangles as BFP approximatiosn are illustrated on four tiles from the test dataset provided for the four aois respectively. Top row: Las Vegas and Paris. Bottom row: Shanghai and Khartoum.

4.2. Extracting Suburban Buildings At Scale

We also conducted an analysis of the building size distribution over 10 major cities in the U.S., and observe that 90% of them have an area smaller than 200 m^2. Buildings of these sizes are typically private homes built in suburban areas. Thus, being able to automatically extract and symbolize suburban houses to the detriment of larger buildings is a worthwhile trade-off in North America. We used this approach to automatically extract 10 million building footprints from VHR satellite images acquired at 50 and 60 cm resolution covering 40 major cities in the U.S and Canada. Examples of this process are given in Fig.4. Given these automatically extracted BFPs, photo interpreters review them and 40% of them get modified or deleted to achieve higher accuracy or to avoid unwanted overlaps. These results match the 75% precision obtained in Las Vegas on the DeepGlobe challenge.

Figure 4. Visual illustration of the proposed automatic BFP extraction and symbolization, in a suburban area of Reno, NV, U.S.A.

5. Conclusion

A CNN architecture to extract and symbolize building footprints from satellite imagery has been proposed. The CNN architecture outputs rotated rectangles providing a symbolized approximation for small buildings. Experiments are conducted on four AOIs, showing best results on suburbs consisting of individual houses. These experiments show that either large buildings or buildings without clear delineation produce weaker results in terms of precision and recall. In future work, this approach could be combined with segmentation-based architectures known to achieve better F1, in order to symbolize larger buildings.

References

[1] DeepGlobe - CVPR2018. `http://deepglobe.org/challenge.html`, 2018. [Online; accessed 01-May-2018].

[2] B. Bischke, P. Helber, J. Folz, D. Borth, and A. Dengel. Multi-task learning for segmentation of building footprints with deep neural networks. *arXiv preprint arXiv:1709.05932*, 2017.

[3] H. Freeman and R. Shapira. Determining the minimum-area encasing rectangle for an arbitrary closed curve. *Communications of the ACM*, 18(7):409–413, 1975.

[4] A. Gruen and P. Agouris. Automatic extraction of man-made objects from aerial and space images. *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps: SMATI 97*, page 228, 1997.

[5] Z. Hayder, X. He, and M. Salzmann. Boundary-aware instance segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, number EPFL-CONF-227439, 2017.

[6] Y. Manolopoulos, A. Nanopoulos, A. N. Papadopoulos, and Y. Theodoridis. *R-trees: Theory and Applications*. Springer Science & Business Media, 2010.

[7] H. Mayer. Automatic object extraction from aerial imagerya survey focusing on buildings. *Computer vision and image understanding*, 74(2):138–149, 1999.

[8] R. Nevatia, C. Lin, and A. Huertas. A system for building detection from aerial images. In *Automatic extraction of man-made objects from aerial and space images (II)*, pages 77–86. Springer, 1997.

[9] J. O'Beirne. Google Map's moat, 2017.

[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[11] J. Redmon and A. Farhadi. Yolo9000: better, faster, stronger. *arXiv preprint*, 2017.

[12] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2017.

[13] M. Rohs, J. Schöning, M. Raubal, G. Essl, and A. Krüger. Map navigation with mobile devices: virtual versus physical movement with and without visual context. In *Proceedings of the 9th international conference on Multimodal interfaces*, pages 146–153. ACM, 2007.

[14] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[15] Uber. Paris: A mobility case study, 2015.

[16] J. Yuan. Automatic building extraction in aerial scenes using convolutional networks. *arXiv preprint arXiv:1602.06564*, 2016.

[17] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai. Multi-oriented text detection with fully convolutional networks. In *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pages 4159–4167. IEEE, 2016.

[18] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang. East: an efficient and accurate scene text detector.