REGULARIZATION OF BUILDING BOUNDARIES IN SATELLITE IMAGES USING ADVERSARIAL AND REGULARIZED LOSSES

Stefano Zorzi and Friedrich Fraundorfer

Institute of Computer Graphics and Vision, Graz University of Technology

ABSTRACT

In this paper we present a method for building boundary refinement and regularization in satellite images using a fully convolutional neural network trained with a combination of adversarial and regularized losses. Compared to a pure Mask R-CNN model, the overall algorithm can achieve equivalent performance in terms of accuracy and completeness. However, unlike Mask R-CNN that produces irregular footprints, our framework generates regularized and visually pleasing building boundaries which are beneficial in many applications.

Index Terms— Generative adversarial networks, building segmentation, boundary refinement, satellite images.

1. INTRODUCTION

Building detection and segmentation from satellite images is still a challenging problem. Automatically detecting constructions and extracting precisely their footprints is in the interest of many engineering and cartographic applications. In recent years, multiple machine learning challenges have been proposed to encourage people to present new building extraction methods (e.g. Deep Globe Challenge[1], SpaceNet Challenge[2], CrowdAI Mapping Challenge[3]).

The most common and effective way to deal with this problem is the use of powerful semantic segmentation or instance segmentation networks. However, in most cases, predicted building footprints have irregular shapes which are very different from the ones used in cartographic applications.

This problem has been recently dealt with Kang et al. [1] where they proposed a building segmentation and refinement pipeline as a solution for the DeepGlobeChallenge 2018. Their framework is composed of a Mask R-CNN [2] model for instance segmentation followed by a boundary refinement algorithm that exploits polygon simplification methods. The overall algorithm produces more realistic building footprints, but it does not consider the intensity image for the regularization to further improve the results.

[1]http://deepglobe.org/challenge.html

[2]https://spacenetchallenge.github.io/

[3]https://www.crowdai.org/challenges/mapping-challenge

In this paper we present a new building segmentation and regularization framework completely based on Deep Learning techniques. The pipeline itself is the same as Kang's, so we still perform the building segmentation as a first step and then we apply the building regularization as a second step. The difference is in the use of a fully convolutional neural network as a regularization method, instead of using polygon simplification algorithms.

Inspired by deep style transfer techniques like pix2pix [3] and cyclegan [4], we train our regularization network using adversarial losses to produce more realistic footprints. In particular, we use OpenStreetMap building footprints as the target footprint domain to train a GAN [5] architecture. We also exploit regularized losses [6, 7] to make the network aware of the real building boundaries in the intensity image and, consequently, to further refine the result. Finally, a reconstruction loss ensures to obtain regularized footprints that look similar in size, pose and shape to the original Mask R-CNN predicted footprints.

The combination of these three types of loss functions enables us to learn a regularization network that not only produces better looking and more realistic building footprints, but is also capable of achieving better scores on the test dataset compared to the pure Mask R-CNN solution.

2. METHOD

Our aim is to learn a mapping function between the domain X (Mask R-CNN footprints) and the domain Y (ideal footprints) given the training samples $\{x_i\}_{i=1}^{N}$ where $x_i \in X$ and $\{y_i\}_{i=1}^{M}$ where $y_i \in Y$. We also exploit RGB images, $\{z_i\}_{i=1}^{N}$ where $z_i \in Z$, to further improve the results training the model with an additional regularized loss.

The model performs the regularization $G : \{X, Z\} \rightarrow Y$ exploiting an encoder-decoder network, as shown in Figure 1. The generation of the regularized footprints is performed by the encoder E_G and the decoder F, so G can be seen as the combination of the two: $G(x, z) = F(E_G(x, z))$. A discriminator D is introduced in order to distinguish between regularized footprints $G(x, z)$ and ideal ones. It is worth noting that the ideal building footprints are not directly evaluated by the discriminator model, but the ideal mask is encoded by E_R and decoded back by the common network F. The aim of this

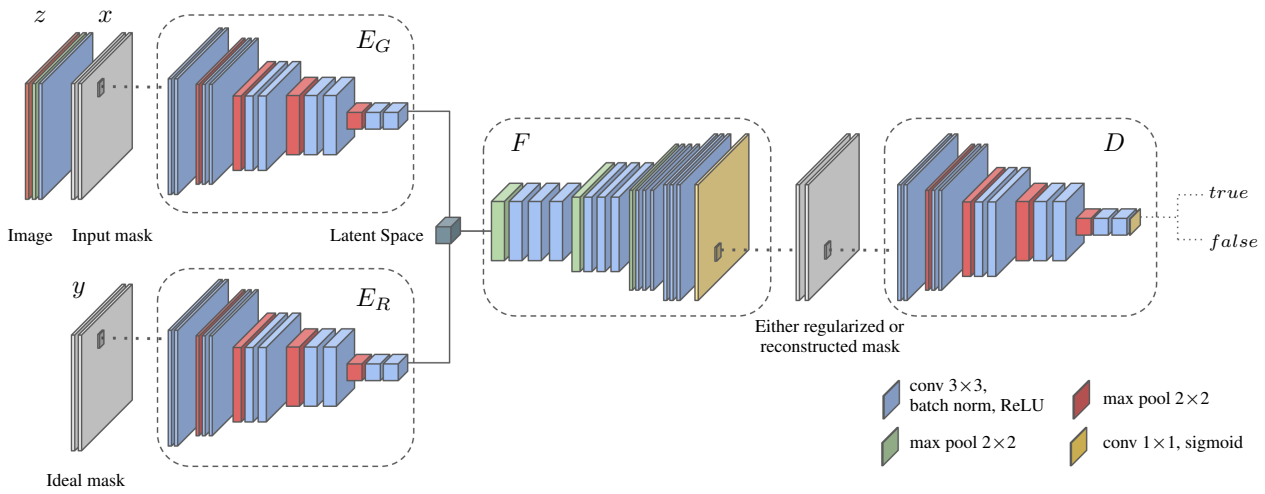

Fig. 1. Workflow of the proposed regularization framework. It is composed of two paths: the generator path ($E_G \rightarrow F$) produces the regularized building footprint mask; the reconstruction path ($E_R \rightarrow F$) encodes and decodes the ideal input mask ensuring to have the same real valued masks as input to the discriminator.

path is to obtain a reconstructed version of y. One concern for this design choice is that the adversarial network can potentially trivially distinguish the two distributions by detecting if the mask consists of zeros and ones (one-hot encoding of the ideal mask), or of real values between zero and one (output of the autoencoder). This problem is solved by generating both reconstructed and regularized samples with the same network F. Also, this architecture ensures stability during training and avoids a winning discriminator situation since the two autoencoders are connected (with the common decoder) and trained together.

The encoders and the decoder are learned exploiting three types of loss functions: *adversarial loss*, *reconstruction losses* and *regularized loss*.

2.1. Adversarial Loss

We use adversarial losses [5] to learn the mapping function between the domain X and Y.

The objective function used to learn the discriminator D is expressed as:

$$\begin{aligned}\mathcal{L}_D(G,R,D) = \mathbb{E}_y[(1 - D(R(y)))^2] \\ + \mathbb{E}_{x,z}[D(G(x,z))^2]\end{aligned} \tag{1}$$

where the path $R(y) = F(E_R(y))$ encodes and reconstructs the ideal mask and the path $G(x,z) = F(E_G(x,z))$ generates building footprints that look similar to ideal footprints in domain Y. The aim of D is to distinguish between regularized footprints and reconstructed footprints. Note that we used the least-squared loss in equation 1 because it ensures better stability during training and generates higher quality results [4].

For the mapping path G the loss function is expressed as:

$$\mathcal{L}_{GAN}(G,D) = \mathbb{E}_{x,z}[(1 - D(G(x,z))^2] \tag{2}$$

This path is trained to fool the discriminator D, in fact, the adversarial loss encourages G to produce footprints similar to the samples on the Y domain.

2.2. Reconstruction Loss

In order to force the network to generate building footprints similar to the input masks, we simply use the *binary cross entropy loss* both on the generator path G and on the reconstruction path R. The loss is computed between x and $G(x)$ and between y and $R(y)$ to produce regularization masks close to the Mask R-CNN predictions and to the ideal masks, respectively. The two losses can be expressed as:

$$\begin{aligned}\mathcal{L}_{BCE_G}(G) &= -\sum_i^N x_i \cdot \log G(x,z)_i \\ \mathcal{L}_{BCE_R}(R) &= -\sum_i^N y_i \cdot \log R(y)_i\end{aligned} \tag{3}$$

2.3. Regularized Loss

Without *regularized losses* our model would not be able to exploit image information to further improve the building regularization.

Alongside the adversarial loss and the reconstruction loss, the *Potts loss* [6] and the *normalized cut loss* [6, 7] are used to learn our model. These two loss functions force the generator G to produce building footprints aligned to the building

boundaries observed in the intensity image. Also, trained with these losses, the generator is capable of solving some artifacts produced by Mask R-CNN (Figure 2).

Potts and normalized cut loss functions can be expressed as:

$$\mathcal{L}_{potts}(G) = \sum_k S^{k\top} W (1 - S^k) \tag{4}$$

$$\mathcal{L}_{ncut}(G) = \sum_k \frac{S^{k\top} \hat{W} (1 - S^k)}{1^\top \hat{W} S^k} \tag{5}$$

where W and $\hat{W}$ are a matrices of pairwise discontinuity costs or *affinity matrices*, while $S = G(x, z)$ is the k-way softmax segmentation mask generated by the network. S^k describes the vectorization of the k-th channel in the segmentation image. In our case $k = 2$ since we have two classes.

2.4. Full Objective

The full objective used to train the generator G and the reconstruction R model is a linear combination between the *adversarial loss*, the *reconstruction loss* and the *regularized loss*.

$$\begin{aligned}\mathcal{L}(G, R, D) = \alpha \mathcal{L}_{GAN}(G, R, D) \\ + \beta \mathcal{L}_{BCE_G}(G) + \gamma \mathcal{L}_{BCE_R}(R) \\ + \delta \mathcal{L}_{Potts}(G) + \epsilon \mathcal{L}_{ncut}(G)\end{aligned} \tag{6}$$

Note that the losses through the paths G and R are obtained switching the encoders E_G and E_R. Once the total loss has been computed, the backpropagation step is performed and the weights of E_G, E_R and F are updated jointly.

3. IMPLEMENTATION DETAILS

3.1. Dataset

We trained our regularization framework on a satellite image which represents the city of Jacksonville, Florida. The image is obtained by performing the pansharpening between the panchromatic layer and three multispectral channels (infrared, green, blue). There is no technical reason why we use the infrared channel. The decision has been taken just for a visualization preference, since grass and trees highlighted in red make the roofs of the buildings more visible to the naked eye. Input masks are generated by a Mask R-CNN model trained using OpenStreetMap footprints. OpenStreetMap footprints are also used as ideal masks during the regularization framework training. In order to achieve better results, our models are learned using single building instances instead of patches. As a test-set, we manually labeled an image of a residential area in Jacksonville mainly composed of mid-sized and small-sized buildings. The size of the test area is around 360×620 squared meters and it counts 243 buildings.

Metric	Recall	Precision	$F_{0.5}$	IoU
Mask R-CNN	0.885	**0.933**	0.923	0.833
Our (no reg. loss)	0.854	0.932	0.916	0.805
Our	**0.909**	0.932	**0.927**	**0.852**

Table 1. Scores of building extraction computed on the test area.

3.2. Network Architecture

The network follows the same design choices of a classical convolutional autoencoder, as shown in Figure 1. The encoders E_G, E_R and the discriminator D share the same architectural design. They are composed of a chain of 3×3 convolutional layers with stride 1, followed by batch normalization layers and 2×2 pooling layers. After every down-sampling operation, height and width of the tensor are halved, but the number of convolutional filters is doubled.

The decoder network F has the dual architecture. It is composed of a chain of 3×3 convolutional, batch normalization and up-sampling layers. This time, after every up-sampling layer, the resolution increases but the number of channels of the tensor is gradually reduced.

3.3. Training Details

For the training every building mask and the corresponding RGB picture are resized to 256×256 pixels images. The ideal masks are generated drawing the OpenStreetMap building footprint polygons in 256×256 pixels masks as well.

For all the experiments we use Adam optimizer with a batch size of 8. The models are trained for 80000 batches in total. All networks are learned from scratch with an initial learning rate of 0.0002. We keep the same learning rates for 60000 batches and linearly decay the rates to zero over the last 20000 batches.

We set $\alpha = 3$, $\beta = 3$, $\gamma = 1$, $\delta = 200$ and $\epsilon = 2$ in Equation 6. ϵ and δ are linearly increased from zero to 2 and 200, respectively, during the first 30000 batches to keep the learning more stable.

The weight matrix W and $\hat{W}$ for *potts loss* and *normalized cut loss* are constructed as:

$$w_{ij} = e^{\frac{-\|F(i)-F(j)\|_2^2}{\sigma_I^2}} \cdot \begin{cases} e^{\frac{-\|X(i)-X(j)\|_2^2}{\sigma_X^2}} & \text{if } \|X(i)-X(j)\|_2 < r \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where $X(i)$ and $F(i)$ are the spatial location and pixel value of node i, respectively. In Equation 7 we use $\sigma_I = 0.075$, $\sigma_X = 4$ and $r = 19$ both for W and $\hat{W}$. Images and masks have values normalized between 0 and 1.

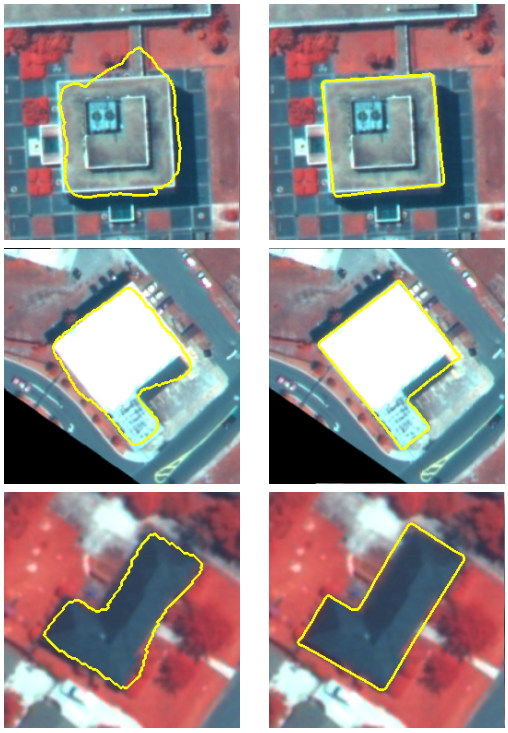

Fig. 2. Comparison between footprints produced by Mask R-CNN (left column) and our regularization method (right column).

4. EXPERIMENTAL RESULTS

The performances of our algorithm are evaluated based on the Intersection over Union (IoU) metric. Computing the scores, we want to analyze the effects of building regularization on the building extraction, comparing the result of the pure Mask R-CNN model with the result of the regularization pipeline (Mask R-CNN and regularization).

Table 1 shows the scores for Mask R-CNN and our method. Although Mask R-CNN shows slightly higher precision values, our regularization pipeline achieves higher results on recall, $F_{0.5}$ and Jaccard index (IoU) scores.

We also trained a model without Potts and normalized cut losses. The scores show higher results for the complete regularization method, a sign that the regularized losses are effective and can be used to refine the segmentation results.

To summarize, our method produces better representations of building footprints with more regular boundaries. Some regularization examples are shown in Figure 2, while a regularized portion of the test area is shown in Figure 3.

5. CONCLUSIONS

We presented a building extraction method that combines a Mask R-CNN model for instance segmentation with a network for footprints regularization. The regularization network has proved capable of exploiting effectively the information of the intensity image to further refine building boundaries, achieving equivalent or even higher results in terms of Intersection over Union compared to the pure Mask R-CNN model. Moreover, unlike Mask R-CNN that produces irregular building masks, our method generates regularized footprints that can be used in many cartographic and engineering applications.

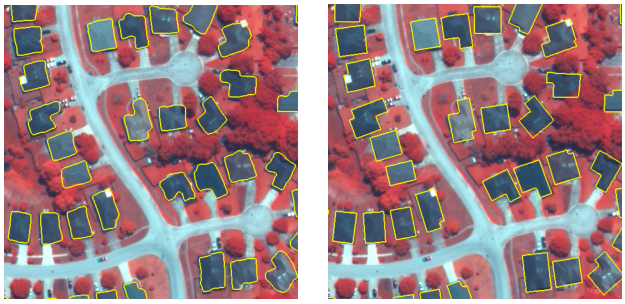

Fig. 3. Portion of the test area evaluated by Mask R-CNN (left) and regularized by our framework (right).

6. REFERENCES

[1] Kang Zhao, Jungwon Kang, Jaewook Jung, and Gunho Sohn, "Building extraction from satellite images using mask R-CNN with building boundary regularization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 247–251.

[2] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, "Mask R-CNN," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2980–2988.

[3] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros, "Image-to-image translation with conditional adversarial networks," *CoRR*, vol. abs/1611.07004, 2016.

[4] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," *arXiv preprint*, 2017.

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[6] Meng Tang, Federico Perazzi, Abdelaziz Djelouah, Ismail Ben Ayed, Christopher Schroers, and Yuri Boykov, "On regularized losses for weakly-supervised CNN segmentation," *arXiv preprint arXiv:1803.09569*, 2018.

[7] Meng Tang, Abdelaziz Djelouah, Federico Perazzi, Yuri Boykov, and Christopher Schroers, "Normalized cut loss for weakly-supervised CNN segmentation," in *IEEE conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City*, 2018.