ENSEMBLE LEARNING AND ENERGY MINIMIZATION FOR ROAD SEGMENTATION ON SATELLITE IMAGES

Domink Alberto, Xiaoao Song, Weiyi Wang, Pavel Pozdnyakov

Group: Nuria

Department of Computer Science, ETH Zurich, Switzerland

ABSTRACT

In this paper, we compared the performance of state-of-art convolutional neural networks on road segmentation from satellite images. We ensembled two Fully Convolutional Network models and two DeepLabv3 models that backboned on Resnet . All of our final models are pretrained on an external data set. To further boost the performance, we implemented a graph cut-based energy minimization algorithm to post-processing the result. With these techniques, our accuracy score can easily reach over 90 percent on Kaggle leaderboard.

1. INTRODUCTION

Semantic Segmentation of an image is a process of labeling every pixel of the image as belonging to a particular class. In the case of the Road Segmentation we are dealing with a binary classification problem. For each pixel of a given image it should be decided, whether it is part of a road, or not.

Motivation. Modern transportation systems grow constantly in their size and complexity. The need for detailed maps for personal and industrial use as well as analysis of these systems in order to improve their efficiency is essential for the economical growth. This includes an automatic update of the information[1] and the detection of damaged roads among others[2]. In the combination with relative ease of acquiring images it makes the problem of Road Segmentation important.

One of the challenges in the problem of Road Segmentation is the fact that the roads on satellite images look visually similar to rivers and railways[3]. One class of models which has shown a good performance for differentiating between such kind of objects are Convolutional Neural Networks (CNNs)[4, 5, 6, 7]. One of the positive characteristics of CNNs for Road Segmentation is their ability to make mostly correct assumptions about the locality of pixel dependencies[8]. Using CNNs for Road Segmentation and Image Segmentation in general exhibits, however, several problems.

One problem is that despite the good pixel-wise predictions the results of CNN predictions are still lacking the visual structure [9]. Another problem is, that in order to achieve good prediction results, a large amount of training data is required[10]. Yet, another problem arises, if we want to use images of different sizes as an input to a CNN, which also affects the efficiency of pre-training[11]. Complex CNNs with many dense layers can be further bounded in their performance by the lack of hardware resources[12].

Contribution. In this paper we first explore the performance of different network architectures on our particular Task. We compare the results of the ResNet[13] with various number of layers, U-Net[14], DeepLabv3[15] and their combinations. We then try several different approaches to tackle the problems arising when using CNN's for Image Segmentation and in particular Road Segmentation mentioned earlier. To make better use of the images of different sizes as well as to make training more memory efficient we use Fully Convolutional Networks (FCNs).

To be able to combat the lack of data, we pre-train our models on the MS-COCO[16] image set. We preprocess the images by padding them and use the context window of the original image size to have more context for each patch we have to make predictions for. We further improve the performance of our models by making an ensemble out of the best performing network architectures we have tried. It happens to be a combination of two FCN and two DeepLabv3 networks[15].

Finally, in order to enhance the semantic visual structure of the predictions produced by our models we introduce a novel approach in the post-processing phase. We make use of the Graph-Cut algorithm[17, 18]. This allows us to deal better with the disturbing and not following the road structure cut-offs of roads, produced by our FCNs.

Related work. There are many papers published on the topic of using CNN for Image Segmentation. One of the earliest is [19]. Over the years different architectural types of CNNs for Image Segmentation were developed. Among them VGG-16[20], ResNet[13], Inception[21] and Xception[22]. Yet another approach, convolution with up-

sampled filters, was used in DeepLabv3[15]. This allowed to enlarge the field of view of filters to incorporate larger context without increasing the number of parameters or the amount of computation[23].

Aerial Image Segmentation was tackled by applying a combination of a FCN with the VGG-16 as a backbone in [24]. Another approach to Road Segmentation was based on the U-Net CNN[14].

In order to deal with the lack of training data several techniques can be applied. One is to generate more training data[24]. Another approach is to use external data sets for pre-training[25]. In this paper we have chosen the second approach and used MS-COCO data set.

Considering the success of the DeepLabV3 and FCNs in general in the tasks of Image Segmentation we have trained a FCN and the DeepLabV3 individually, as well as made an ensemble out of them. We then make use of our idea to add a Graph-Cut algorithm as a post-processing step to our best scoring models in order to achieve the better visual structure for our output, which is in our opinion the most important contribution of this paper.

2. MODELS AND METHODS

Problem Statement. Given a dataset $\mathcal{D}$ consisting of RGB satellite images and road segmentation masks pairs $\left(x^{(i)}, y^{(i)}\right) \in \mathcal{D}$, we aim to design an algorithm that will assign a binary label $z \in \{0, 1\}$ to the pixels of an unseen RGB image x, depending on whether they belong to the road or to the background. The prediction accuracy score will be used as a performance measure for the models. In the given training data, there is a problem of class imbalance, i.e., from visual inspection of the images it follows that there is more background than roads. The proportion of background pixels to road pixels is unknown. The training set consists of 100 RGB satellite images of size 400×400 pixels with 100 corresponding ground-truth maps of the same size. The test set consists of 94 RGB images of size 608×608.

Baseline Models. As our baseline models we used the two models provided by the course as a part of the exercise 9. The first one uses the mean of the RGB values, one for each channel, and their variances to perform the prediction. The prediction is performed patch-wise with the patch size of 16×16 pixels. To train the model Logistic Regression is used. On the positive side of this model is its simplicity and fast and efficient training from the perspective of performance and memory usage. The downside of this model is the low accuracy of predictions. When trained on the whole data set it produced the accuracy of approximately 0.508 on the public test set.

The second baseline model is CNN with the following structure. As in the case with the first baseline model, it operates on the patches of size 16×16 with 3 channels, red, green and blue. Each patch is then fed to a convolutional layer of the depth of 32 with the filter window size of 5×5 an the stride of 1. The convolutional layer is followed by a max pooling layer of size 2 with stride 2. Then another convolutional layer comes again, now with the depth of 64 and the same filter window size and stride as the first one, 5×5 and 1 respectively. Then max pooling layer comes, again with the size of 2 and the stride of 2. The two fully connected layers of the depth 512 and 2 respectively are at the end of the network. The softmax function is then applied to the output of the last layer.

For the weights the L2 regularization is used. The optimizer is the Momentum Optimizer with an exponential decay. The decay rate is 0.95. The initial learning rate is 0.01. The activation function for convolutional layers is RELU. The activation function for the fully connected layers is the identity. This model produces similar accuracy as the first baseline model when trained for five epochs without the pre- or post-processing of the training set, 0.52 on the public test set.

For both models the whole patch is predicted to be a part of the road if and only if the percentage of pixels of that patch labeled as belonging to a road lies above 25%. Both models also share the same weakness. The patch size of 16×16 is too small to give enough contextual information on whether it belongs to the road or to the background.

From CNNs to Ensemble of FCNs. One major drawback of the two baseline models is the use of patches which are too small to contain enough contextual information for good predictions. However, even when using the patches of a bigger size, the predictions fail on the frontiers between the roads and the background[26]. This is again mainly due to the loss of the contextual information in the final network layers in particular[26]. In order to preserve as much context as possible, our first natural idea was, therefore, to introduce a context window as large as possible around the patch. Hence, as a pre-processing step we introduce a padding around the images and use the context window of the size of the whole image, 400×400 in our case. This did not make much difference in comparison to the results of the second base model.

Keeping the dimension of the input, context window with the 16×16 patch inside it at 400×400 we train a known CNN architecture and evaluate its performance. We have chosen ResNet[13]. We train several different versions of ResNet varying the depth of the network, 18, 34, 50 and 101 respectively. The prediction results are improved, achieving in average an accuracy of approximately 0.8891 on the public test set.

We then pre-train our models on the MS-COCO image set. Since the test images as well as the ones used for pre-training have different dimensions as the train im-

ages, our next step is to find a way for a network to be able to make good predictions on images of different sizes. We have, therefore, trained a FCN, Deeplabv3[15] and their ensemble. Better results produced by FCNs in comparison to CNNs for images of different sizes follows from the fact that, when we use dense layers in CNN, the size of the resulting weight matrix is dependent on both the size of the layer and the size of the previous layer, since each neuron in the previous layer is connected to each neuron in the dense layer. This is obviously not the case for FCNs.

Graph-Cut. To further improve the accuracy of our model, we applied interactive segmentation using a graph cut method [18]. The idea behind this post-processing step is to overcome the rather sudden cut-offs of roads produced by our FCNs, not following the road structure even though qualitatively no pixel color change is observed. Consider the following weighted graph $G = (V, E)$:

- Set V contains all n pixels and two sentinel nodes A and $\bar{A}$, indicating "road" and "not road" respectively.
- Each pixel is connected to A with an edge weight of $\tilde{y}^s$, and to $\bar{A}$ with an edge weight of $1 - \tilde{y}^s$
- Adjacent pixels are connected with an edge weight proportional to their similarity in the LAB color space.

The binary mask prediction z can be derived by finding a minimum cut separating A and $\bar{A}$, then assigning pixels according to their partition. The graph cut-based energy function can be defined as follows:

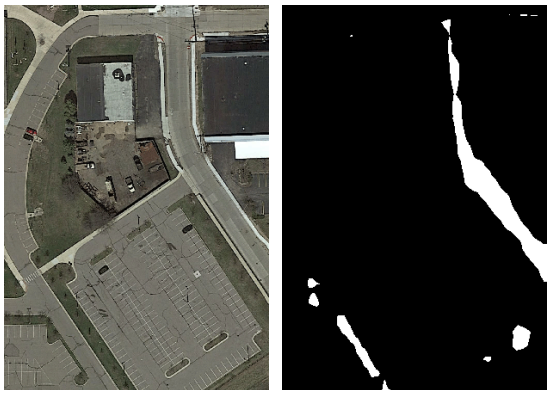
$$E(Z) = \sum_{i=1}^{n} \psi_i(z_i) + \lambda \sum_{i=1}^{n} \sum_{j \in N(i)} \phi_{i,j}(z_i, z_j) \quad (1)$$

where λ is the hyperparameter, $z_i \in \{0, 1\}$ is the binary label of the i^{th} pixel, $\psi_i(0) = 1 - \tilde{y}_i^s, \psi_i(1) = \tilde{y}_i^s$, and

$$\phi_{i,j}(z_i, z_j) = \begin{cases} \exp\left(-\frac{(I_i - I_j)^2}{2\sigma^2}\right) & \text{if } z_i \neq z_j \\ 0 & \text{otherwise} \end{cases}$$

I_i and I_j are 3 -dimensional LAB color values for the i^{th} and j^{th} pixels. The function $\phi_{i,j}$ penalizes heavily for similar pixel colors when $|I_i - I_j| < \sigma$, while being close to zero when adjacent pixels have very different colors.

If we compare the original image in 1(a) with the Ensemble prediction in 1(b), we notice that the road is thinned out in the upper right part of the image and the upper left part of the image isn't recognized as road at all. This is a problem we can solve with graph cut by controlling the λ. The coefficient λ specifies the relative importance of the local properties versus the boundary properties. In this case, the colors of road and background is very similar. This enforces us to choose a rather small λ, as we don't apply Graph Cut segmentation to discover any new roads, but rather follow and enhance the structure of existing ones to get continuous and smooth road predictions.



(a) Original image (b) Ensemble

(c) Graph Cut ($\lambda = 0.1$) (d) Graph Cut ($\lambda = 0$)

Fig. 1. Example of bad prediction by Ensemble that can be improved with Graph Cut using the right λ.

3. EXPERIMENTAL RESULTS

Experimental setup (Data Set and Evaluation Metrics). We did all the experimenting with a dataset consisting of 100 RGB images with dimension 400 and corresponding 0-1-groundtruth masks. All images are randomly rotated and horizontally and vertically flipped with a chance of 50 percent. Their brightness, contrast and saturation was randomly changed slightly. A random crop of the image is taken, resized to 400×400 and since our models are pretrained with the MS-COCO set, we subtract and scale with the associated mean and standard deviation.

We evaluate the performance of our models by observing the prediction accuracy, which is measured by the fraction of correctly predicted patches.

Non-pretrained models Results (Comparison).

In Table 1. the prediction accuracy scores for the different versions of ResNets and the U-Net are depicted. They outperform the two baseline models drastically. Noticeably, the deeper the models are the less accurate is their prediction.

Pretrained models Results (Comparison). The results

for our pretrained models can be seen in Table 3. There's a further improvement for every model compared to its non-pretrained counterpart model. Interestingly, in case of pre-trained models we observe that the deeper ResNet101 model performs better than the shallower ResNet50 model. Our best score, with a prediction accuracy of 0.905, was achieved by the Ensemble method. Intuitively, one might question that why there are no comparison between ResNet18(and ResNet34) and Deeplabv3. This is because Pytorch doesn't currently provide the ResNet18(and ResNet34) backbone for segmentation models. Though it would need a few changes to BasicBlock in order for it to start to be possible, it is not the focus of our paper.

Graph Cut Results (Comparison). The choice of λ is crucial for Graph Cut post-processing. We therefore carried out grid search to determine the optimal value. We started with $\lambda = 10$ and gradually decreased the parameter until at $\lambda = 10^{-4}$ the amount of newly labeled pixels became insignificant compared to the previous λ and we jumped to $\lambda = 0$, which produced the best results. In Figure 1 it's visually clearly observable that a greater λ better follows the road structure, as we can see the outline of the whole road in Figure 1(c). However, the trade-off is that background pixels with similar colors initiate new road segments. Choosing $\lambda = 0$ omits such behaviour, and only enhances existing road segments. On the other hand, small disturbances in the pixels colors challenge the spreading more. Figure 1(d) shows an example where Graph Cut with $\lambda = 0$ is actually able to enhance the road structure partially. This is a rare case. Most images spread beyond the road structure into nearby objects. We've quantitatively evaluated the different λ parameters on all models and show the results for our best performing ensemble method in Figure 2. This pattern of underperforming, even in the case of $\lambda = 0$ represents all models we've tried and not only shows the specific case. The Graph Cut algorithm is not able to improve the overall accuracy of our predictions. Compared to the baselines, it still outperforms these methods by far.

Method (not pretrained)	**Accuracy**
ResNet-18	0.887
ResNet-34	0.876
ResNet-50	0.872
ResNet-101	0.870
ResNet-152	0.869
U-Net	0.866

Table 1. Train by different depth of ResNet model (without pretrained weights) used against achieved score on public leaderboard on Kaggle.

λ	10	1	.1	.01	.001	.0001	0
Acc	.66	.72	.85	.897	.899	.900	.901

Table 2. With Graph Cut post-processed predictions of the Ensemble method against the achieved public score on Kaggle.

Method (all pretrained)	**Accuracy**	
	w/o Graph Cut	w/ Graph Cut
ResNet-50	0.897	0.889
ResNet-101	0.903	0.897
Deeplabv3 (ResNet-50)	0.897	0.891
Deeplabv3 (ResNet-101)	0.894	0.886
Ensemble	**0.905**	0.898

Table 3. Different methods used against achieved score on public leaderboard on Kaggle. The left accuracy column contains the results directly from the FCNs. The right accuracy column shows the results after post processing the FCN outputs with Graph Cut.

4. SUMMARY

In this paper we examined several different approaches for the problem of Road Segmentation. We saw that using patches to train a model and make predictions does not produce satisfactory results, even if the context window around the patch is sufficiently large. Being able to use the information from the whole image, however, in combination with FCNs produced good results. Even for the roads partially covered with other objects, the predictions were rather accurate. This is also true for differentiating between roads and similarly looking objects, e.g, rivers, the problem discussed in[3].

Although we have expected that such elaborated architectures as DeepLabv3 are able to significantly outperform other, generally simpler, models we have tried, there was rather little improvement. However, we were able to achieve good results using combinations of DeepLabv3 with those simpler networks. Our proposed enhancement of road structures using the post-processing step with Graph-Cut also did not bring much improvement. Mainly because the semantic context of the image can't be grasped by the energy function and the trade-off of having more correctly classified road pixels against having more falsely classified road pixels when choosing λ doesn't allow good results. As a possible next steps to further improve the prediction performance, we see the development of newer pre- and post-processing techniques, as well as the use of large and more diverse data sets for training.

5. REFERENCES

[1] Volodymyr Mnih and Geoffrey E. Hinton, "Learning to detect roads in high-resolution aerial images," in *Computer Vision – ECCV 2010*, Kostas Daniilidis, Petros Maragos, and Nikos Paragios, Eds., Berlin, Heidelberg, 2010, pp. 210–223, Springer Berlin Heidelberg.

[2] H. Ma, N. Lu, L. Ge, Q. Li, X. You, and X. Li, "Automatic road damage detection using high-resolution satellite images and road maps," in *2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS*, 2013, pp. 3718–3721.

[3] Corentin Henry, Seyedmajid Azimi, and Nina Merkle, "Road segmentation in sar satellite images with deep fully convolutional neural networks," *IEEE Geoscience and Remote Sensing Letters*, 02 2018.

[4] Srinivas Turaga, Joseph Murray, Viren Jain, Fabian Roth, Moritz Helmstaedter, Kevin Briggman, Winfried Denk, and Hyunjune Seung, "Convolutional networks can learn to generate affinity graphs for image segmentation," *Neural computation*, vol. 22, pp. 511–38, 11 2009.

[5] Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," 01 2009, p. 77.

[6] Yann Lecun, Fu Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," 01 2004, vol. 2, pp. II–97.

[7] Kevin Jarrett, Koray Kavukcuoglu, Marc'Aurelio Ranzato, and Yann Lecun, "What is the best multi-stage architecture for object recognition?," 09 2009, vol. 12.

[8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton, "Imagenet classification with deep convolutional neural networks," *Neural Information Processing Systems*, vol. 25, 01 2012.

[9] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," *CoRR. arXiv*, 12 2014.

[10] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele, "The cityscapes dataset for semantic urban scene understanding," 06 2016.

[11] Mahdi Hashemi, "Enlarging smaller images before inputting into convolutional neural network: zero-padding vs. interpolation," *Journal of Big Data*, vol. 6, 12 2019.

[12] K. Siu, D. M. Stuart, M. Mahmoud, and A. Moshovos, "Memory requirements for convolutional neural network hardware accelerators," in *2018 IEEE International Symposium on Workload Characterization (IISWC)*, 2018, pp. 111–121.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," 06 2016, pp. 770–778.

[14] Zhengxin Zhang and Qingjie Liu, "Road extraction by deep residual u-net," *IEEE Geoscience and Remote Sensing Letters*, vol. PP, 11 2017.

[15] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam, "Rethinking atrous convolution for semantic image segmentation," 06 2017.

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, Eds., Cham, 2014, pp. 740–755, Springer International Publishing.

[17] Yuri Boykov and Vladimir Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sept. 2004.

[18] Yuri Y Boykov and M-P Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *Proceedings eighth IEEE international conference on computer vision. ICCV 2001*. IEEE, 2001, vol. 1, pp. 105–112.

[19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton, "Imagenet classification with deep convolutional neural networks," *Neural Information Processing Systems*, vol. 25, 01 2012.

[20] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv 1409.1556*, 09 2014.

[21] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," 06 2015, pp. 1–9.

[22] Francois Chollet, "Xception: Deep learning with depthwise separable convolutions," 07 2017, pp. 1800–1807.

[23] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, 06 2016.

[24] Pascal Kaiser, Jan Wegner, Aurelien Lucchi, Martin Jaggi, Thomas Hofmann, and Konrad Schindler, "Learning aerial image segmentation from online maps," *IEEE Transactions on Geoscience and Remote Sensing*, vol. PP, pp. 1–15, 07 2017.

[25] Matthew D. Zeiler and Rob Fergus, "Visualizing and understanding convolutional networks," in *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, Eds., Cham, 2014, pp. 818–833, Springer International Publishing.

[26] Taibou Sekou, Moncef Hidane, Julien Olivier, and Hubert Cardot, "From patch to image segmentation using fully convolutional networks - application to retinal images," 04 2019.

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Ensemble Learning and Energy Minimization for Road Segmentation on Satellite Images

Authored by (in block letters):

For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**
Song	Xiaoao
Pozdnyakov	Pavel
Alberto	Dominik
Weiyi	Wang

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date

Zürich, 31.07.2020

Signature(s)

Xiaoao Song

D. Alberto

Weiyi Wang

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.