

Machine Learning - Project

Road Segmentation

Marion Chabrier, Valentin Margraf, Octavianus Sinaga
Department of Computer Science, EPFL Lausanne, Switzerland

Abstract—The goal of this project is to implement and train a Neural Network that is able to segment satellite images into roads and background. We implement a Convolutional Neural Network and use the Sliding Window Approach in order to tackle this classification task. TO BE ADDED: RESULTS

I. INTRODUCTION

Image segmentation is a well known task in the field of Deep Learning which can be handled by Neural Networks. There exist different approaches such as Convolutional Neural Networks ([1]) or newer ones such as the U-Net ([2]) which both yield good results according to our research.

We decided to choose the Convolutional Neural Network since it seemed very natural to us to classify pixels of the image by taken into account their context inside the image.

In the first section we explain our methods i.e. data augmentation, our model, training of the model as well as tuning of the hyperparameters. We then present our results obtained with the CNN. We obtain TO BE ADDED good results

II. METHODS

A. Classification Task

Our Neural Net takes as input parts of the image of size 72x72, so called windows, and classifies if the 16x16 patch in the center of this window belongs to "road" or not. If the net classifies the patch as road, the output should be [0,1]. Hence it will be white in the predicted label. Otherwise it should be [1,0] and the patch in the label will be black. This is illustrated in Figure 1.

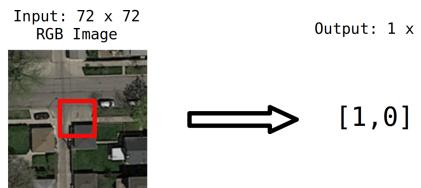


Figure 1: Our CNN classifies if the patch in the center of the window belongs to road or to background.

The idea behind this approach is, that the patch in the center is classified by taken into consideration the surrounding of this patch. E.g. if we have a lot of vehicles near or in the patch, it is more likely that the patch belongs to road. The patch hence is somehow classified by its context in the image.

B. Data Augmentation

The training data consists of 100 RGB satellite images of size 400x400 and their groundtruth (400x400 binary image) respectively. Obviously this task is not easy. Sometimes the roads are hidden by a tree or cars may drive on them. We therefore augment our training data in order to achieve better results. We rotate each image and its groundtruth by 15, 30, 45, 60, 90, 100, 180 and 270 degrees. When rotating we obtain bigger images, where several parts are just black. We therefore fill those parts by reflecting the image along its boundary axis. Then we crop out images of size 400x400 to get training images and groundtruths in the same size as the 'original' training images, groundtruths respectively.

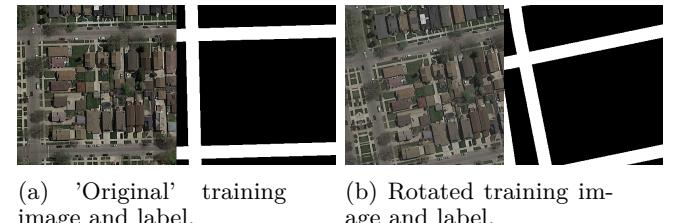


Figure 2: Example of augmenting data: Rotated by 15 degrees, mirrored along the boundary axes and cropped out 400 x 400 pixels.

By doing so we augmented the amount of data by a factor of 9.

C. Convolutional Neural Net

With aim of classifying the whole image we move our window through the whole image. This so called 'sliding window' approach is illustrated in Figure 3.

In order to apply this technique to patches near the border, we have to enlarge the images. We therefore



Figure 3: For each window the patch in the center will be classified. We then move the window through the image by a certain stride.

choose a padding method: At the border we reflect each image along its boundary axes.

SAY SOMETHING ABOUT WE USED TENSORFLOW... The architecture of our Convolutional Neural Net is displayed in Table 1. TO BE ADDED CHOICE OF LAYERS, DROPOUT

Layer	Characteristics
Input	72x72x3 RGB
Conv + LReLU	64 filter each 5x5
Max Pooling	2x2
Dropout	0.25
Conv + LReLU	128 filter each 3x3
Max Pool	2x2
Dropout	0.25
Conv + LReLU	256 filter each 3x3
Max Pool	2x2
Dropout	0.25
Dense + LReLU	128 nodes
Dropout	0.5
Softmax + Output	2 labels

Table I: Architecture of the Convolutional Neural Network.

D. Training

In the training process we randomly choose a window of an image of the training data. We then compute the corresponding patch of the groundtruth image. The groundtruth label is then given by the mean of all pixels in this patch: If it is higher than a certain threshold, 0.25 in our case, then the label will be black, otherwise white.

By doing so we get even much more training data since each window of each image corresponds to one training image. But since some of the windows overlap themselves, we don't gain much more information by using all of them.

TO BE ADDED. TUNED HYPERPARAMETERS: ALPHA, DROPOUT, ACTIVATION FUNCTIONS , WINDOW/PATCH SIZE!?! WHY WE CHOSE THEM IN THIS MANNER takes a lot of time 3 000 000 parameters to be trained GPU BLABLA

We chose the patch size to be 16, because submit on ai crowd!! window not too big because of computational cost

CROSSVALIDATION???

CROSSVALIDATION???

CROSSVALIDATION???

CROSSVALIDATION???

E. Prediction

The test image data consists of 50 RGB satellite images of size 608x608. We first have to do some data preparation: We enlarge the images by a padding method similar to the one explained in section II-C. After cropping the image into windows of size 72x72, we feed each window into our neural network and predict a label for the patch in the center of the window. We get our final predictions by mapping each label back to its corresponding patch and setting these patches together to an image. We use a stride that has the same size as our patch. Thus we don't miss too much information, since the windows overlap.

III. RESULTS

For every test image we obtained a prediction, which segments roads from background. Two example images and their corresponding groundtruth are displayed below.



Figure 4: Test image 2 and its prediction.



Figure 6: Test image and its predictions, patch size 16 and 4 respectively.



Figure 5: Test image 9 and its prediction.

As one can observe, a patch size of 16x16 and window size of 72x72 gives not very smooth predictions of road. This is obviously caused by the fact, that we predict one label for a whole patch. If we choose smaller patches, we get smoother result.

IV. DISCUSSION

V. SUMMARY

ACKNOWLEDGEMENTS

REFERENCES

- [1] S. Bittel, V. Kaiser, M. Teichmann, and M. Thoma, “Pixel-wise segmentation of street with neural networks,” 2015, <https://arxiv.org/pdf/1511.00513.pdf>.
- [2] O. Ronneborger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” 2015, <https://arxiv.org/abs/1505.04597>.