Satellite Image Segmentation with Convolutional Neural Networks

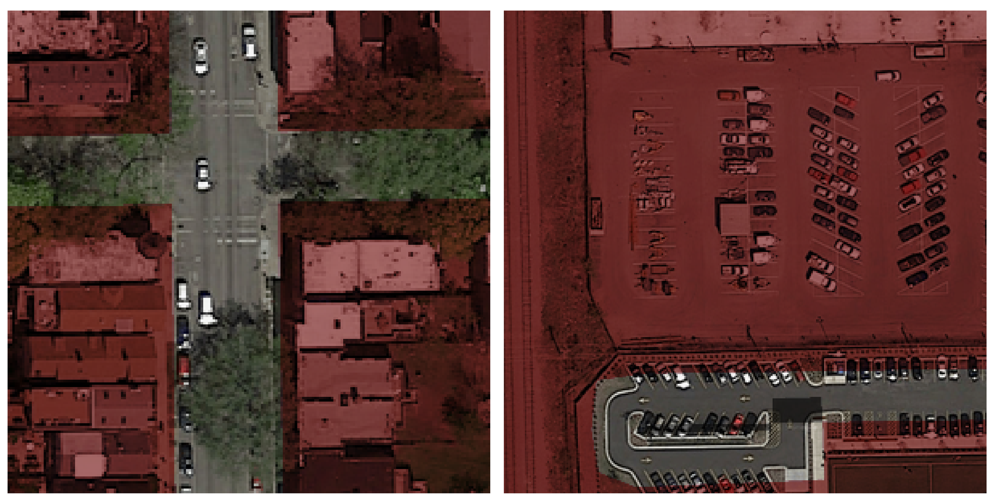
Dario Pavllo dario.pavllo@epfl.ch
Mattia Martinelli mattia.martinelli@epfl.ch
Chanhee Hwang chanhee.hwang@epfl.ch

École polytechnique fédérale de Lausanne

Introduction

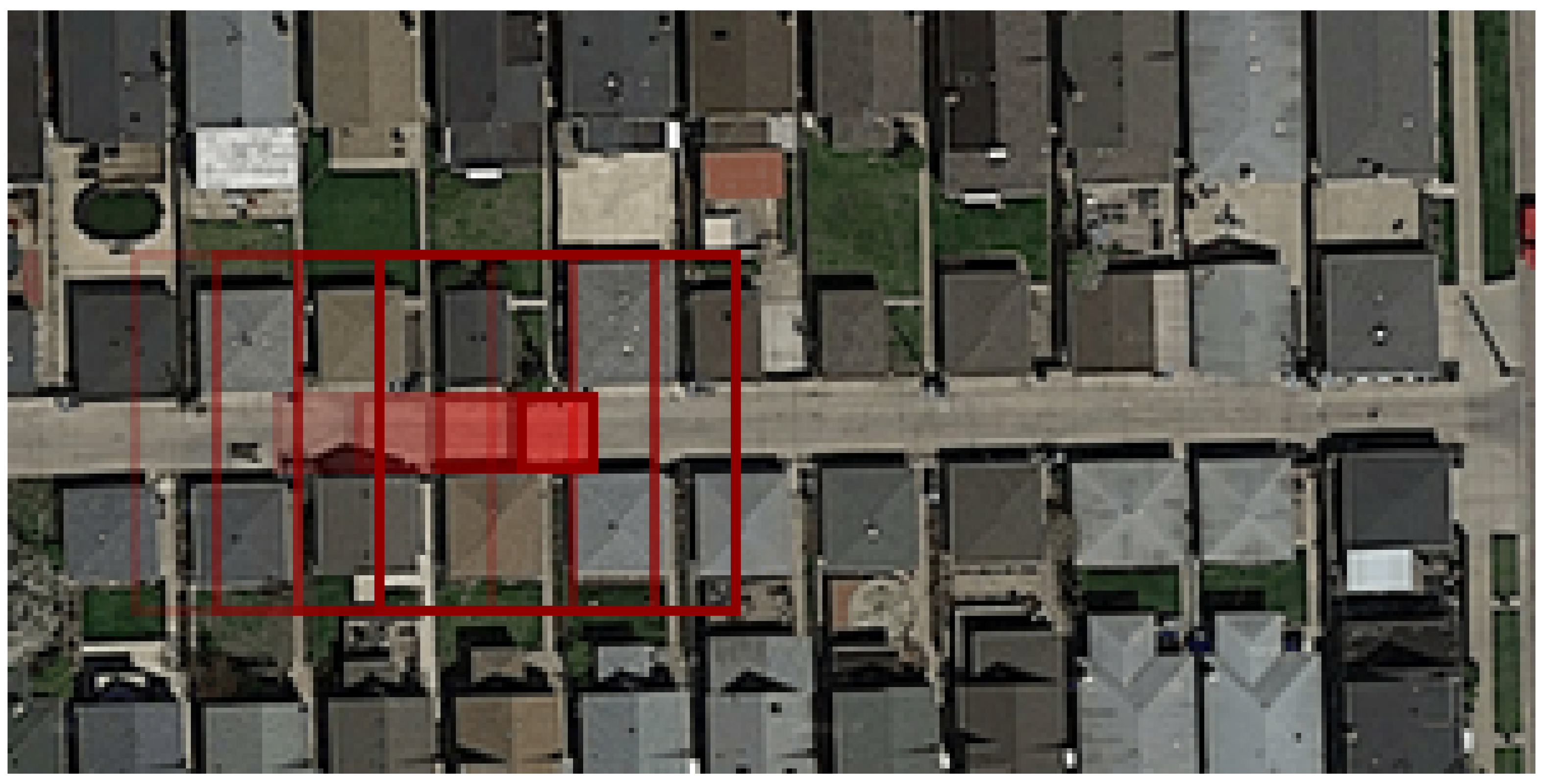
Image segmentation is a research topic that is becoming increasingly popular for various tasks in computer vision, especially for what concerns the extraction of features from satellite images. In particular, this project aims to build a classifier that is able to detect whether an area represents a road or not, and label each pixel of the images accordingly. The task poses some challenges, as some roads are not directly visible (for instance, because they are covered by trees), and the labelling of the training set might be inconsistent in certain cases (e.g. parking lots and walkways).

A custom **convolutional neural network (CNN)** has been developed in order to solve this specific task, and has proved to generate correct predictions with an accuracy of 93.6% on the test set. This score allowed us to reach the first place on the Kaggle competition.



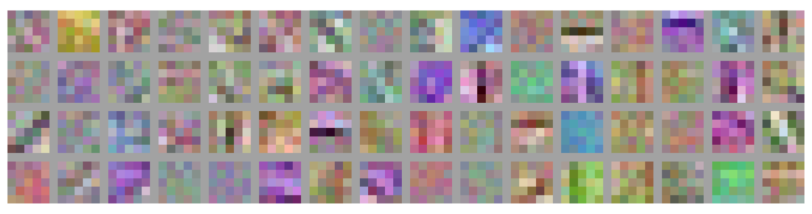
Methodology

- **Sliding window approach:** the objective is to classify the block at the centre of an image according a certain context, which corresponds to a small square window that is iterated on each image of the training set. The optimal size has been shown to be 72×72.



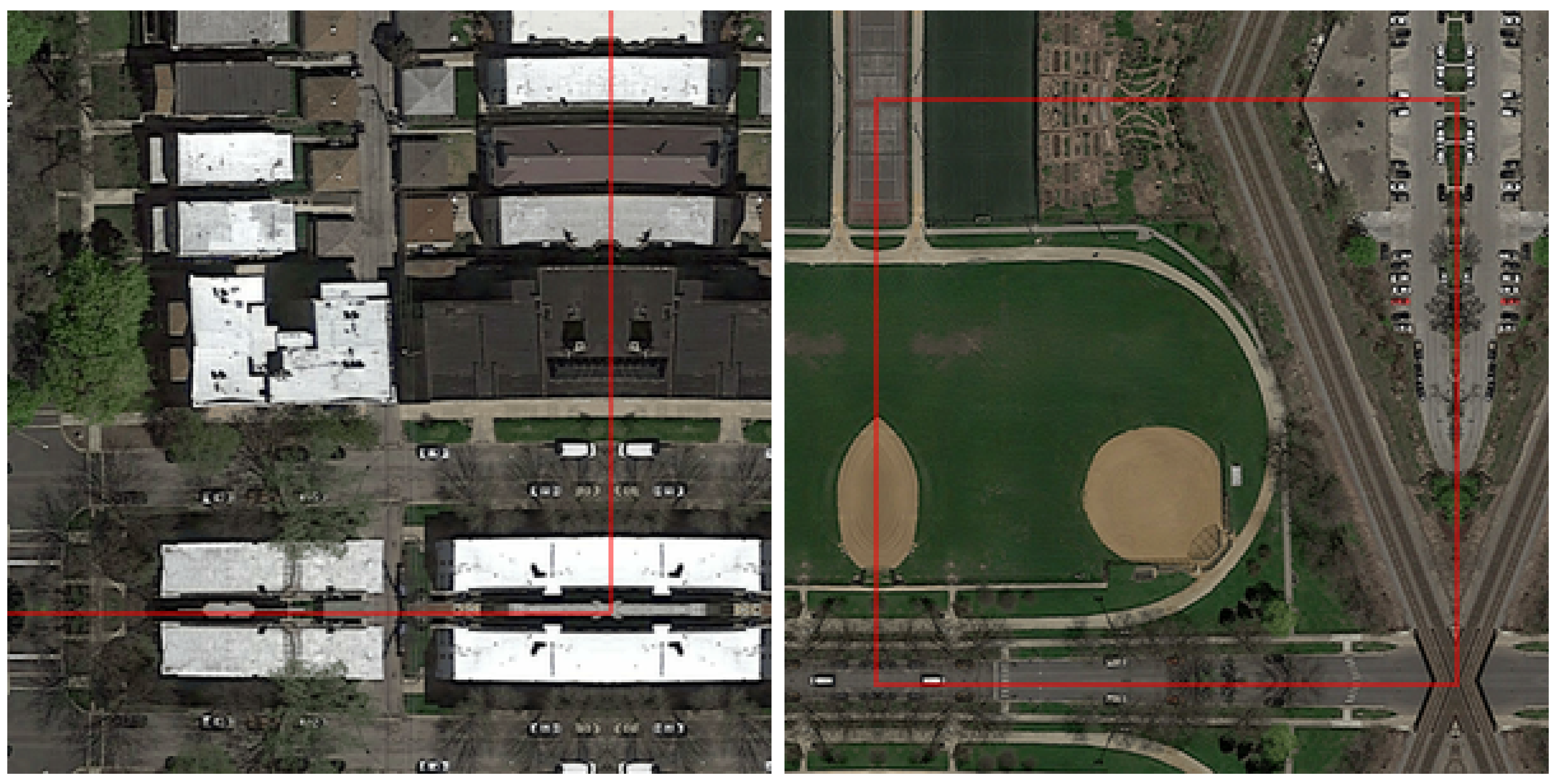
- **Leaky ReLU activation functions:** various experiments on our dataset have revealed that plain ReLU activations cause the so-called *dead filters*, thereby deteriorating the accuracy of the model. The Leaky ReLU function with a high leak factor $\alpha = 0.1$ has solved this problem and has improved the classification performance.

Dead filters



Good filters

- **Real-time image augmentation:** each sample (i.e window) is randomly rotated in steps of 90 degrees and flipped horizontally/vertically during the training procedure, thereby yielding an 8x increase of the dataset size.
- **Regularization:** in order to reduce overfitting, dropout layers are added after every max-pooling layer. Furthermore, L2 regularization is used on the weights of the fully connected layers.
- **Handling of boundaries:** prior to training or classification, the images are extended using mirror boundary conditions. This produces a good estimate of their missing content and allows the algorithm to predict the correct label at image borders.



Implementation

- The model has been implemented with *Keras* library (along with *Theano* backend) and has been trained on a GPU, using the Adam optimizer (an efficient variant of stochastic gradient descent) to minimize the cross-entropy.
- When taking into account every possible window position and every permutation caused by the image augmentation, the memory requirement for the training set becomes prohibitive (in the order of terabytes). For this reason, a real-time training set generator has been implemented: it runs in a dedicated thread, and it generates mini-batches of samples using a Monte Carlo method. Each mini-batch consists of a set of randomly sampled (and transformed) windows.
- The table shows the complete structure of the neural network:

#	Type	Notes
1	Input	$72 \times 72 \times 3$
2	Convolution + Leaky ReLU	64 5×5 filters
3	Max Pooling	2×2
4	Dropout	$p = 0.25$
5	Convolution + Leaky ReLU	128 3×3 filters
6	Max Pooling	2×2
7	Dropout	$p = 0.25$
8	Convolution + Leaky ReLU	256 3×3 filters
9	Max Pooling	2×2
10	Dropout	$p = 0.25$
11	Convolution + Leaky ReLU	256 3×3 filters
12	Max Pooling	2×2
13	Dropout	$p = 0.25$
14	Fully connected + Leaky ReLU	128 neurons
15	Dropout	$p = 0.5$
16	Output + Softmax	2 neurons

Results

- During the internal experiments, the model has shown a validation accuracy of $92.89\% \pm 0.7\%$ (using a 4-fold cross-validation).
- The final test, evaluated on the hidden test set of the Kaggle competition, obtained an accuracy of 93.6%.
- As can be seen from the pictures, despite some glitches, the model segments the images in a human-like manner, and predicts the correct label even in the case of obstruction by trees.