Where is a Waldo?
A deep learning approach to template matching

Thomas H[3] Hossler, Department of Geological Sciences, Stanford University.

Introduction

Template matching is a well-known computer vision challenge where an algorithm is trying to find similarities between two or more different images [2]. These methods are slow and often lack accuracy. A Deep-Learning approach to the template matching challenge is proposed here. The objectives are: 1) prove that the deep learning method works, 2) shows that it can beat computer vision algorithms such as SIFT [3] and 3) try to create a method where template relaxation is possible (matching the template to some extent).

Methods

Two Convolutional Neural Networks will be simultaneously trained. One on the template image and one the query image. The first ConvNet will be used as an hypernetwork [1], outputing a kernel *K* of weights. This kernel is reshaped and used for the final convulational layer of the other ConvNet. The ouput is a binary image that will be compared to true location of the template.

Dataset

The algorithm will be tested on a brick-and-mortar retail store dataset. An example is presented Figure 1. Each image contains several items who are listed with their coordinates. The dataset contains over 86,000 items from different stores. Most of the items (template) appears only once in the dataset.

Network architecture

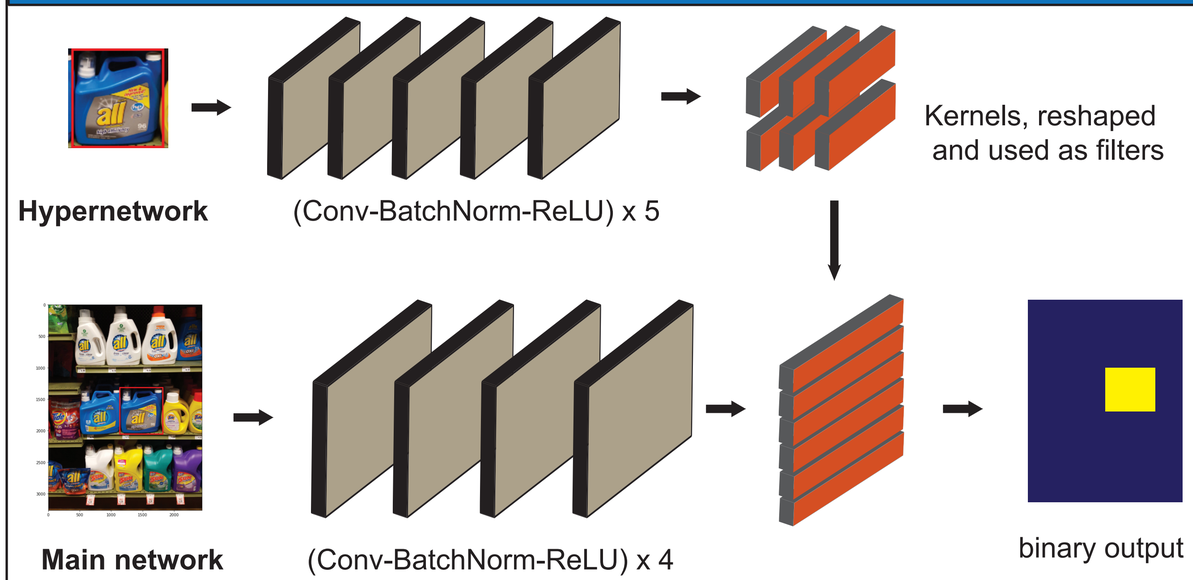
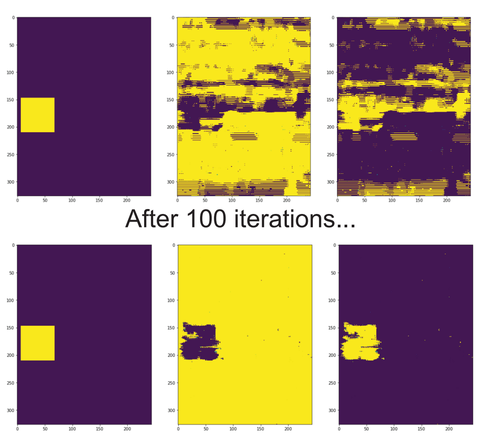


Figure 1: Architecture of the ConvNet created. The hypernetwork is made of 15 layers and the main network is made of 12 layers. The output of the hypernetwork is reshaped and used as weights for the last layer of the main network. A softmax activation function is used to create a binary output and a Negative Log Likelihood loss function is used. An Adam optimizer with scheduled annealing and L2 regularization has been chosen.

Overfitting one template



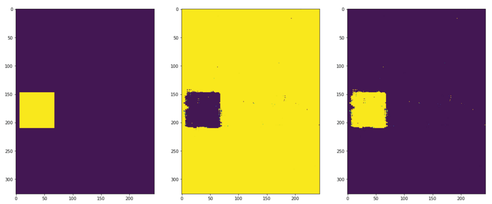


Figure 2: Output of the ConvNet after the softmax activation. The true location of the template is presented on the left. The two channels of the output - probability that a given is equal to zero or one - are shown after 100, 600 and 1,000 iterations (middle and right). The ouput is slowly fitting the “truth” while the loss converges to zero.

Conclusion

The first steps to create a template matching algorithm have been successfully made. Using an hypernetwork, the author has shown that the information contained in the template image can be carried to the main network and be used to perform template recogniton.

Next steps

In a second time, the performance of the algorithm will be studied under the following variations:

- size of the kernel *K*
- depth and width of the hypernetwork
- depth and width of the main network

Pretrained ConvNet, such as VGG, will be used to speed up the learning process. Finally, this method will be compared to more traditional computer vision techniques to see if an improvement in accuracy is observed.

References

[1] D. Ha, A. Dai, and Q. V. Le. HyperNetworks. 2016.
[2] N. S. Hashemi, R. B. Aghdam, A. S. B. Ghiasi, and P. Fatemi. Template Matching Advances and Applications in Image Analysis. 2016.
[3]D. G. Lowe. Object recognition from local scale-invariant features. In Computer vision, 1999.

Acknowledgements

The author would like to thank Focal Systems for providing the data as well as the GPU necessary to run the algorithm.