A Benchmark on Deep-Learning Convolutional Neural Network for the Representation of Natural Scenes with Large Seasonal Variations

Amandine Gout[1,3]

Yann Lifchitz[1,3]

Titouan Cottencin[1,3]

Quentin Groshens[1,3]

Shane Griffith[2,3,4]

Jérémy Fix[1,4]
jeremy.fix@centralesupelec.fr

Cédric Pradalier[2,3,4]
cedric.pradalier@georgiatech-metz.fr

[1] Centrale Supelec,
Metz, France

[2] College of Computing,
Georgia Institute of Technology,
Atlanta, USA

[3] GeorgiaTech Lorraine,
Metz, France

[4] CNRS UMI 2958 GT-CNRS,
Metz, France

Abstract

This document demonstrates the format requirements for papers submitted to the British Machine Vision Conference. The format is designed for easy on-screen reading, and to print well at one or two pages per sheet. Additional features include: pop-up annotations for citations [**? ?**]; a margin ruler for reviewing; and a greatly simplified way of entering multiple authors and institutions.

All authors are encouraged to read this document, even if you have written many papers before. As well as a description of the format, the document contains many instructions relating to formatting problems and errors that are common even in the work of authors who *have* written many papers before.

1 Introduction

Check `bmvc_guidelines.pdf` for the original text.

2 Related Works

- Deep nets for plant recognition [5],

- Deep nets for place recognition [7],
- Change detection across seasons, [3], not based on deep nets
- the framework of cedric and shane for the lake dataset and image alignements : "Survey Registration For Long-Term Natural Environment Monitoring"
- Oxford article on regression, [4]
- AlexNet paper with great details on parameter choice, [2]
- OxfordNet, [6]
- Average faces over American Yearbooks, [1]

3 Methodology

3.1 Classification

Labels chosen to divide the dataset in 1-meter intervals around the lake, along with heading separated in 10 degrees increments. Goal : obtain classes of images representing the same area with the same angle, to facilitate classification.

At training : class selection and reindexing. All classes must have at least 50% of the highest class image count. This way, all classes are represented enough times for the network to learn their features. This represents around 1k images per class, for around 300 classes.

Fine tuning was made on number of images (more is better), size of minibatches (more is faster, less is more precise) and learning rate (decrease more times, but less each time). Other parameters remained as is.

The aim was not necessarily to obtain a great classification, but to create good features for each class, independently of seasonal changes. We want good convolutional layers.

Implementation made on Caffe, which is a good framework for classification tasks. No changes on the network structure, but trained from scratch. Unlike regression, Caffe was adapted to the task.

3.1.1 Prototypes

Representations of an image will be found in the network's convolutional layers. As such, we averaged filter responses for every layer over all images in a class. We can thus minimize the impact of variance over the class images (slight heading and position variations).

3.2 Regression

Labels: - We need the label to include the information on the position and angle of the view - 4 dimensions labels (coordinates of the robot + coordinates of the point we are looking at on the shore) - Normalization issues: The solution was to recenter and rescale the labels in [0,1]. - Datasets creations: Same as for the classification we built our datasets with images from the most highly represented classes. By taking a given number of images from each highly represented class we make sure that no class is overly represented so that we do not induce bias.

Loss: - Euclidean loss, made possible because our labels were homogeneous

Network structure: - CaffeNet (a replication of the model described in the AlexNet publication) that performs well for classification tasks (on the ImageNet dataset). - Accuracy layer is irrelevant on a regression problem so we removed it (the loss is our accuracy indicator) - We changed the loss layer to the euclidean loss layer

Implementation: - Framework Caffe: easy to use for the usual cases, python wrapper - Not directly adapted for our problem: float labels, multilabel regression - Can work with some tricks - We would not recommend Caffe for similar problems

4 Dataset and Experimental Setup

Dataset: - Environmental images from 2013 to 2015 - Seasonal changes - Lighting changes - Video frames from camera XXXXX

Classification:

Regression: - Gaussian initialization of weights had to be tuned so that the convolution layers pass information to the fully connected layers (variances and biases) - Learning rate adaptation depending of the nature of the layer (fc or conv). - Weight decay - Tests after loading the weights from the classification - Influence of the dropout layer

5 Results

5.1 Classification

Our best training on the aforementioned dataset was made with 300 000 iterations over mini-batches of size 32, with a learning rate decreasing 10 times throughout training. The full training took 5 days to finish on a Tesla K20C machine, and attained 70% accuracy on network predictions, on a top-1 classification basis. Such results are satisfactory considering the resemblance of natural scene images. It should be noted that classification was not the main goal, but a good classification will intuitively lead to better class representations.

We can thus extract the trained filters responses to see images as processed by the network. The conv1 layer will mostly learn edges, namely the skyline and the waterline. Conv2 detects foliage, and convolutional layers 3 through 5 contain low-level features that are harder to interpret properly. We tested prototype generation on all convolutional layers, as well as the last pooling layer pool5.

We generated prototypes (some results still pending, will modify when tested) for all classes, and tested whether an image can be recognized only using its class prototypes. Testing over a thousand random images and measuring using a cosine distance shows that all layers perform well for classification. [WILL INCLUDE STATS LIKE MEDIAN DISTANCE] [WILL INCLUDE SOME GRAPHS FOR THIS]

We also trained the same network on half the classes, to test for generalization capabilities. We observed the same results as before on known classes, and found that pool5 is just as good on unknown classes. The pool5 layer is thus a great choice of seasonal-invariant representation, being both compact and able to generalize. [MORE GRAPHS TO COME]

5.2 Regression

6 Conclusions

References

[1] Shiry Ginosar, Kate Rakelly, Sarah Sachs, Brian Yin, and Alexei A Efros. A century of portraits: A visual historical record of american high school yearbooks. *arXiv preprint arXiv:1511.02575*, 2015.

[2] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. URL http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-ne pdf.

[3] Peer Neubert, Niko Sünderhauf, and Peter Protzel. Appearance change prediction for long-term navigation across seasons. In *ECMR*, pages 198–203. IEEE, 2013. ISBN 978-1-4799-0263-7.

[4] Tomas Pfister, Karen Simonyan, James Charles, and Andrew Zisserman. Deep Convolutional Neural Networks for Efficient Pose Estimation in Gesture Videos. In Daniel Cremers, Ian D. Reid, Hideo Saito, and Ming-Hsuan Yang, editors, *ACCV (1)*, volume 9003 of *Lecture Notes in Computer Science*, pages 538–552. Springer, 2014. ISBN 978-3-319-16864-7.

[5] Angie K. Reyes, Juan C. Caicedo, and Jorge E. Camargo. Fine-tuning Deep Convolutional Networks for Plant Recognition. In Linda Cappellato, Nicola Ferro, Gareth J. F. Jones, and Eric SanJuan, editors, *CLEF (Working Notes)*, volume 1391 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2015.

[6] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[7] Niko Sünderhauf, Feras Dayoub, Sareh Shirazi, Ben Upcroft, and Michael Milford. On the Performance of ConvNet Features for Place Recognition. *CoRR*, abs/1501.04158, 2015.