Object tracking with deep neural networks

Santeri Salmijärvi

School of Electrical Engineering

Bachelor's thesis
Espoo Work in progress! Compiled: 12:50:45 2017/06/25

Thesis supervisor:

D.Sc. (Tech) Pekka Forsman

Thesis advisor:

M.Sc. (Tech) Mikko Vihlman

Aalto University
School of Electrical
Engineering

AALTO UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING

ABSTRACT OF THE
BACHELOR'S THESIS

Author: Santeri Salmijärvi

Title: Object tracking with deep neural networks

Date: Work in progress! Compiled: 12:50:45 2017/06/25 Language: English
Number of pages: 5+7

Degree programme: Bachelor's Program in Electrical Engineering

Supervisor: D.Sc. (Tech) Pekka Forsman

Advisor: M.Sc. (Tech) Mikko Vihlman

abstract in english

Keywords: keywords in english

AALTO-YLIOPISTO
SÄHKÖTEKNIIKAN KORKEAKOULU

KANDIDAATINTYÖN
TIIVISTELMÄ

Tekijä: Santeri Salmijärvi

Työn nimi: Kohteenseuranta syvillä neuroverkoilla

Päivämäärä: Work in progress! Compiled: 12:50:45 2017/06/25 Kieli: Englanti Sivumäärä: 5+7

Koulutusohjelma: Sähkötekniikan kandidaattiohjelma

Vastuuopettaja: TkT Pekka Forsman

Työn ohjaaja: DI Mikko Vihlman

lyhyt tiivistelmä suomeksi

Avainsanat: avainsanat suomeksi

Contents

Abbreviations

CNN Convolutional Neural Network

DNN Deep Neural Network

MLP MultiLayer Perceptron

NN Neural Network

ReLU Rectified Linear Unit

1 Introduction

Object tracking is a large and actively researched sub-area of computer vision. The main task for a tracker is to find and follow the desired subject in a sequence of images. Object tracking is closely related to other image analysis tasks so the implementations also share elements. In the recent years, research has been made in using deep neural networks for object tracking.

Many of the deep networks tailored to tracking tasks are variations of convolutional networks

Comment by author:
add cites to examples

. They consist of visible input and output layers with several so-called hidden layers in between them. Another way of extracting features from a frame is stacked denoising autoencoder [1]. The training of deep neural networks requires a large amount of training data and their development has been made easier by an increase in the size of applicaple datasets.

The goal of this thesis is to study the concepts of object tracking and deep neural networks. It will present the architectures and principles currently used in deep neural networks tailored to object tracking tasks. The practices and datasets used in training and evaluating such networks are also introduced.

2 Deep Learning

This chapter introduces the basic concepts and is based on the book Deep Learning by Goodfellow et. al. [2]

2.1 Deep neural networks

A Deep Neural Network (DNN) is commonly defined as a Neural Network (NN), that has a **visible** input and output layer with several **hidden layers** between them. The distinction between visible and hidden layers is important because training of the network only evaluates the output layer's performance. During training, a **learning algorithm** optimizes the individual hidden layers to best approximate the desired output of the whole network.

The input layer takes in the data to be processed, which typically means a vector of color values in the case of object tracking. These are then processed by the hidden layers and finally the output layer produces the target's position in the frame. These models usually come in the form of a **feedforward neural network** or **MultiLayer Perceptron (MLP)**. The name comes from the fact that information flows from the input through computations to the output with no **feedback** connections.

In NNs, each layer consist of several **units** with a weight and activation function. A bias-term can also be defined for each unit. The weights of a layer are commonly represented by a matrix by which the input-vector is multiplied. Units in a layer also have a common activation function. A unit's activation function is fed by the sum of its weighted inputs in addition to the possible bias, and the result is output to the next layer alongside the layers other units' outputs. A commonly used unit type is the **Rectified Linear Unit (ReLU)**, which is defined by the activation function $g(z) = \max\{0, z\}$. It provides a nonlinear transformation while being comparable to linear models in terms of generalizing well and being easy to optimize.

Comment by author:
picture from eg. deeplearningbook page 174?

Before training, the weights of a MLP are initialized to small random values and biases to zero or small positive values. Then an algorithm called **stochastic gradient descent** is commonly applied alongside a training dataset. The basic procedure is to calculate the error of the netwok's output values compared to the desired ones using a **loss function**. The function's gradient can then be calculated for example by **back-porpagation**, which feeds the errors back through the network to assign a contribution value to each unit. These values are then used to calculate the gradient of the loss function relative to the weights. Each weight is adjusted slightly to the opposite sign to minimize the loss function.

2.2 Convolutional networks

A Convolutional Neural Network (CNN) is simply a NN that uses convolution instead of general matrix multiplication in at least one of its layers. The main benefits of convolution in NNs are that it's dramatically more efficient in terms of memory requirements, it reduces the amount of computation needed and it makes it possible to work with variable input sizes.

A typical convolutional layer consists of three stages: a convolution stage, detector stage and pooling stage. These can be implemented by individual layers. First, a **kernel** is applied to the input data in positions separated by a stepsize. This means that a linear activation function is fed by the matrix product of the input location and the kernel's weight matrix. In the detector stage, the results are then run through a non-linear activation, for example a ReLU. Finally, a **pooling function** is used to combine the results of multiple nearby outputs as the final output.

Comment by author:
pictures from dl p.337?

Convolutional layers enable indirect connections to all or most of the input data deeper in the network even when individual layers' connections are very sparse.

3 Object tracking

Object tracking in video sequences has been researched for decades and NN-based algorithms have recently been researched Comment by author: cite works here as NNs' capabilities in classification tasks have been noted Comment by author: cite relevant works There has also been development in the area of benchmarking the tracking algorithms [3].

3.1 Overview

Tracking methods can be roughly divided to generative and discriminative, but combinations of them have also been proposed.

Generative methods search the frame for the best matches to a template of an appearance model of the subject. Template methods based on pixel intensity and color histograms perform well with no drastic changes in object appearance and non-cluttered backgrounds. Appearance models learnt from training can be less affected by appearance variations and adaptive schemes provide added flexibility, while sparse models handle occlusion and image noise better. [4] Comment by author: add citations

Discriminative methods consider tracking as a binary classification problem. They take the background also into account to separate the target from it. Used approaches include refining the initial guess with a support vector machine [5] or utilizing a relevance vector machine [6]. Comment by author: go into greater detail about the methods

3.2 Deep neural networks in tracking

(Overview of task from DNN-point of view, strengths and weaknesses compared to more traditional solutions.)

An early implementation of a CNN-based tracker [7] pre-dates the work of Krizhevsky et. al. [8]. It takes modifies the architecture used for detection to make the network less affected by sifts in the objects position in the frame. Shift-invariancy is a non-desirable quality in tracking while using previous positions as a as it might result to mixups with objects similar to the target. [7]

3.3 Data sets and evaluation

Overview of the data sets used for training and analysis. Methods used for comparing performance.

4 Challenges

Subsections dealing with the present challenges in object tracking and examples for dealing with them.

5 Conclusions

Summarize the current state of object tracking with DNNs with possibly some insight to future developments.

References

[1] Wang, Naiyan and Dit Yan Yeung: *Learning a deep compact image representation for visual tracking*. In Burges, C. J. C., L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (editors): *Advances in Neural Information Processing Systems 26*, pages 809–817. Curran Associates, Inc., 2013, ISBN 9781632660244.

[2] Goodfellow, I., Y. Bengio, and A. Courville: *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[3] Wu, Y., J. Lim, and M. H Yang: *Object tracking benchmark*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 37(9):1834–1848, 2015.

[4] Wang, Q., F. Chen, W. Xu, and M. H Yang: *Object tracking via partial least squares analysis*. IEEE Transactions on Image Processing, 21(10):4454–4465, 2012.

[5] Avidan, S.: *Support vector tracking*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(8):1064–1072, 2004.

[6] Williams, O., A. Blake, and R. Cipolla: *Sparse bayesian learning for efficient visual tracking*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(8):1292–1304, 2005.

[7] Fan, J., W. Xu, Y. Wu, and Y. Gong: *Human tracking using convolutional neural networks*. IEEE Transactions on Neural Networks, 21(10):1610–1623, 2010.

[8] Krizhevsky, A., I. Sutskever, and G. E. Hinton: *Imagenet classification with deep convolutional neural networks*. In *Advances in Neural Information Processing Systems*, volume 2, pages 1097–1105, 2012. http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.