Visual Tracking with Deep Networks: A Performance Study

David Concha, Francisco J. García, Raúl Cabido, Alfredo Cuesta-Infante, Juan J. Pantrigo, Antonio S. Montemayor
{david.concha, raul.cabido, alfredo.cuesta, juanjose.pantrigo, antonio.sanz} @urjc.es, fj.garciae@alumnos.urjc.es

Universidad Rey Juan Carlos
CAPO Computación de Altas Prestaciones y Optimización
NVIDIA GPU EDUCATION CENTER
GOBIERNO DE ESPAÑA MINISTERIO DE ECONOMIA Y COMPETITIVIDAD

Introduction

Deep convolutional networks have gained a lot of attention for object classification and detection problems in the computer vision community. For static images it performs very well when having enough training data. However, for dynamic scenes (videos or image sequences) an exhaustive evaluation for localizing interesting objects with a sliding window approach may involve important computational workloads. Other strategies may introduce the temporal information into the neural network, like in more complex architectures of recurrent neural networks, while penalizing the training stage.

In this work we propose a tracking system using the power of a deep convolutional network for the object detection stage, and guiding the classification stage to relevant zones in future time steps. Moreover, we perform an important performance evaluation using different computing platforms such as a Tegra K1 board, a mobile GPU and CPU and a desktop CPU and GPU.

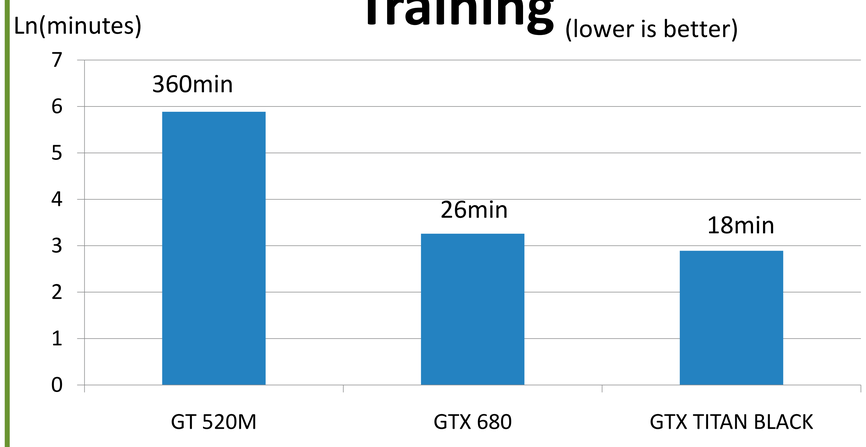
Training

We use the Digits 2.0 framework for training (only on GPU) the convnet with a standard LeNet architecture and the Daimler Mono[1] pedestrian benchmark data set of 640x480 pixels with the following image distribution:

Images	No pedestrian	Pedestrian
Training (88,9%)	353880	12528
Validation and Test (11,1%)	44235	1566

With the network trained, we export the model to Caffe, letting us execute the convnet in different computing platforms (CPU and GPU).

The following graph shows the training time for 10 epochs with different GPUs.



[1] M. Enzweiler and D. M. Gavrila. “Monocular Pedestrian Detection: Survey and Experiments”. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.31, no.12, pp.2179-2195, 2009

Visual tracking

The tracking algorithm works as follows:

-Initially, a sliding window approach is performed to detect a pedestrian (a). The neural network is used to evaluate (classify) each region, and the best one above a given threshold becomes the state estimate (a person detection).

-Once the pedestrian is detected, an fixed local search is performed around a small neighborhood of the previous estimation (b). Finally, the best region is evaluated by the neural network becomes the state estimate for this frame.

(a) Sliding window

(b) Local Search

Frame #2

Frame #10

Frame #20

Performance study

The convnet evaluation is divided in batches of 50 regions of 48x96 pixels each.

The neighborhood needed to classify can grow because the pedestrian can come close to the camera and the region of interest increases and so does the number of batches we need to send to the convnet.

Since this measure can vary from frame to frame, to make a better comparison among different hardware configurations, we report the time needed for each one to evaluate only one batch.

Results show important differences between CPU and GPU performance, but also a feasible tracking execution with much higher framerates than 30 fps.