Homework 4: Face Identification

Prof. Davi Geiger

Due May 16^{th}, 2017

Introduction

We will be using the data set from

http://vis-www.cs.umass.edu/fddb/

and the documentation is at

http://vis-www.cs.umass.edu/fddb/fddb.pdf

and in particular (as discussed in the appendix B, Data Formats) they provide for each image the ellipse parameters $(r_a, r_b, \theta, c_x, c_y)$ for each face detection of their data base. We will use such data for the training the network.

As an environment for learning, I suggest to use Tensor Flow (released by Google) and the site is

https://www.tensorflow.org/

and as specific architecture we will use the Deep Residual Network described in

https://arxiv.org/pdf/1512.03385.pdf

and pre-trained versions exist in Tensor Flow

https://github.com/tensorflow/models/blob/master/slim/README.md#Pretrained

Preparing the data set for training

Select 5 images from the data set. The choice is your criteria (I think with faces facing the camera are simplest and faces not too close to the lenses, so they are not too large and with one face is easier than more faces).

1. Show the 5 images on your laptop.

For each image produce two images of it, one which is a copy of it, and the other which is a blurred ($\sigma = 3$) and decimated version by a factor of 2. So if the original image is of size $M \times N$, the second image will be of size $M/2 \times N/2$, four times less pixels than the original.

2. Show the new 5 smaller images on your laptop.

We will work with fixed size inputs to the neural network. Let us fix the size to be 32×32 pixels. Faces can be identified reliable at a minimum size 16×16 pixels. Since the faces from the data set vary in size, we expect that for some images a window of size 32×32 will be too small to capture the entire face. The faces in the original image are described by a box of size $2 \times (r_a, r_b)$ and centered at (c_x, c_y). Thus, for larger faces where $2 \times \max(r_a, r_b) >> 32$ we will consider the smaller image version of it (of size $M/2 \times N/2$) where the window of fixed size 32×32 will be a better fit to the face.

So you can now create the input to the neural network, for the learning phase. For each of the five (5) images (say size $M \times N$), extract the face data, i.e., all the ellipses in the image, where each ellipse is described by $e_{a,b} = (r_a, r_b, \theta, c_x, c_y)$.

If $2 \times \max(r_a, r_b) \approx 32$ we create forty nine 32×32 windows of faces, each one centered at $d_{mn} = (c_x \pm 2\,m, c_y \pm 2\,n); m, n = 0, 1, 2, 3..$ There are all together forty nine (49) different windows for each face in the image capturing partial face detection. They are all considered face examples with a score given by the measure

$$s_{m,n,a,b} = \frac{d_{mn} \cap e_{a,b} \text{ pixels}}{d_{mn} \cup e_{a,b} \text{ pixels}}$$

Other windows that do not include the faces are considered non-face, the score is zero. Create as many windows that are non-faces as windows that are faces, i.e.,forty nine (49) e non-face images per face in the original image.

If $2 \times \max(r_a, r_b) >> 32$, than consider the smaller image of size $M/2 \times N/2$. For these smaller images $e_{a/2,b/2} = (\frac{r_a}{2\sqrt{2}}, \frac{r_b}{2\sqrt{2}}, \theta, \frac{c_x}{2}, \frac{c_y}{2})$. So perform the same procedure as before, for the smaller image. Check if $2 \times \max(\frac{r_a}{2\sqrt{2}}, \frac{r_b}{2\sqrt{2}}) \approx 32 \quad \rightarrow \quad \max(r_a, r_b) \approx 64\sqrt{2}$. Then, if yes, create twenty five 32×32 windows, centered at $d_{m/2,n/2} = (\frac{c_x}{2} \pm 2\,m, \frac{c_y}{2} \pm 2\,n); m, n = 0, 1, 2$. Notice that there are fewer translations, as we are working on a smaller size image. There are all together twenty five different windows for each face in the image capturing partial face detection. They are all considered face examples with a score given by the measure

$$s_{m,n,a,b} = \frac{d_{m/2,n/2} \cap e_{a/2,b/2}}{e_{m/2,n/2} \cup l_{a/2,b/2}}$$

Other windows that do not include the faces are considered non-face, score is zero. We create as many windows that are non-faces as windows that are faces, i.e., twenty

five non-face images per face in the original image.

3. Show three different face windows per image and the associated score.

Summary: We now should have many input windows of size 32×32 to feed the neural network. Each window will have associated to it a score. If it is not a face, the score is zero. Some of these windows came from an image of smaller size (blured and decimated) and other windows came from the original images. For each "face" image we have one non face image extracted from the same image. Note that since for each face we have a few examples of faces (with different scores) as we translate the window, we will have to create as many non-face images.