Chris Sutton

HW #3

1 New Method for Building a Large Face Dataset with Little Annotation

This paper uses a novel method for building a large face data set. As a first step the movie database IMDb is used to obtain a list of mostly actors which is then checked against the free based knowledge graph to produce a rank list of 2500 males and 2500 females. From this list of candidates 200 images are downloaded per celebrity from Google image search. The next stage uses filtering to remove candidates from the list who do not have enough distinct images. Next human annotators are presented with 200 images per candidate, 50 images per page, and asked to retain a celebrity if the set of 200 images is 90% pure. Next, names appearing in LFW and YTF databases are removed to make training against those benchmarks possible. Now with 2622 celebrity names remaining, the search engines Google and Bing are used with the keyword “actor” used to obtain roughly 2000 images per identity. For these 2000 images, a support vector machine is trained to rank the top 1000 images per candidate for retaining in a data set. In the final stages of constructing the large face data set, near duplicates are removed when they only differ in color balance or with text superimposed and finally CNN and SoftMax scores are used to rank the image in each identity. These final ranked images are presented to human annotators and blocks the 200 who are asked to validate a particular block good if it's purity is greater than 95%.

2 New Effective Network Architecture

2.1 Baseline CNN learning- bootstrapping

This paper uses an initial face classifier in the form of a CNN that is trained on N=2622 unique identities that were obtained from the prior section. The CNN solve the N-way classification problem by using a SoftMax log loss on the final fully connected layer (N-Dimensional output) to obtain a score vector for each training image of dimension R^N.

$W\phi(I_t) + b$ is in R^N , W is in $R^{N \times D}$, $\phi(I_t)$ is in R^D

SoftMax Log-Loss

$$E(\phi) = -\sum_t \log\left(e^{\langle \mathbf{e}_{c_t}, \mathbf{x}_t \rangle} / \sum_{q=1,\ldots,N} e^{\langle \mathbf{e}_q, \mathbf{x}_t \rangle} \right)$$

The CNN architecture is designed as a model containing 37 layers, or 11 modules,

which include nonlinear layers after each convolutional layer.

Once the CNN architecture is trained, the weight matrix and the bias from the classification layer could be tossed and the vectors $\phi(l_t)$ could be used to measure the distance between vectors and hence learn faces. However, performance can be significantly improved by using the vectors X in a triplet loss scheme.

2.2 Triplet Loss for Model Improvement

The output of the CNN is L2 normalized with a weight matrix W' having dimensions R^{LxD}. This weight matrix serves as a dimensionality reduction on the output of the CNN. The triplet loss is shown below.

$$E(W') = \sum_{(a,p,n)\in T} \max\{0, \alpha - \|\mathbf{x}_a - \mathbf{x}_n\|_2^2 + \|\mathbf{x}_a - \mathbf{x}_p\|_2^2\}$$

The triple loss uses and anchor face and another face from the positive class of the anchor face along with a third phase which is from the negative class of the anchor face. Above (a,p,n) represent the anchor face the positive class face and the negative class face.

2.3 Training CNN -A

During training images are cropped from a 256 by 256 image to yield a 224 by 224 pixel patches. As the training proceeds the location of the cropped image from the base image is changed every time an image is sampled. Additionally, base images are reflected left to right with the 50% probability. Training is by stochastic gradient descent using mini batches of 64 samples and momentum coefficient of .9. Additionally, drop out and weight decay is used.

2.4 Training CNN -B, D

Training configurations B&D are done by adding fully connected layers to CNN-A then initializing the layers and training with a lower learning rate.

2.5 Testing

Add test time, the embedding descriptors are compared by Euclidean distance to see if the distance is smaller than a threshold Tau. The paper also talks about 2D alignment of the test images helping performance as shown in table 4.