Kin Recognition Using Weighted Graph Embeddings - Progress Report

Manu Varma : mv465@cam.ac.uk
Supervisor : Daniel Bates
Director of Studies : Robert Mullins
Overseers : Frank Stajano and Amanda Prorok

1 Work Accomplished

Given a pair of images of faces and a given kin relationship (ex. Father-Son), I wanted to be able to tell whether the people in the images had that relationship. To do this, my solution was to implement the paper, "Weighted Graph Embedding-Based Metric Learning for Kinship Verification".

I was able to implement the main algorithm, WGEML, from the paper. This included creating the functions that would get the necessary face descriptors, Local Binary Patterns, Histogram of Gradients, Scale-Invariant Feature Transform and a face descriptor created from the VGG convolutional neural network, from each image. WGEML would be done for each type of kinship relationship and would take, as input, any positive pairs of that kinship relationship and any negative pairs, where each pair would have all of the face descriptors created for each image in the pair. It would then output a weight and a matrix for each face descriptor that was used. Thus, the prediction step was also implemented using these matrices and weights as well as any data preparation functions that were necessary and the project was run end-to-end.

The accuracies that were obtained were within 5% of what the paper, which satisfies my success criterion of replicating the paper's mean verification accuracies within a 15% error range. However, for one of the datasets, TSKinFace, no negative pairs were used to test the algorithm at first so I had to create my own negative pairs for TSKinFace and, when it was used, it dropped the accuracies of the dataset by about 8%, on average. The average accuracy of the dataset dropped from 92% to 84%. Although this is still within the error range that I gave myself in my success criterion, this led me to believe that the original paper didn't end up testing TSKinFace on any negative pairs, as I had very similar results to the paper before I added the negative pairs. Lastly, ablation studies on the face descriptors were run for each dataset.

Furthermore, I implemented unit tests along the way to make sure that my functions were running as I expected them to run, which leads me to believe that I properly implemented the main algorithm. I managed to get a code coverage of 97% for my unit tests, which excludes any scripts that were written.

2 Schedule and Unexpected Difficulties

The project is currently on schedule. The main unexpected difficulty occurred from the fact that the paper I was implementing was vague in certain parts of the project. For example, one of the outputs of the main algorithm was a matrix called U_p for each face descriptor used and the dimensions of such were $D \times d$ where D was the dimension of the face descriptor. However, $d \ll D$ was all that was given for d. Thus, the value of d had to be experimented on to find a good value for it. In the end, since all of the face descriptors had their dimensions reduced to 200 in the paper, I ended up choosing a value of $d = 10$ since the maximum accuracy the TSKinFace dataset would occur when $d = 2$ and for the KinFaceW datasets, it would occur at $d = 21$ so setting $d = 10$ was a good compromise between the two.