CIS 520 Final Project Report
On gender and age classification of facial images

Gokul Subramanian
gosub@seas.upenn.edu

Kripa Iyer
akripa@seas.upenn.edu

December 3, 2010

1. Introduction

The process of gender and age classification of facial images has been studied extensively in the field of machine learning and computer vision. In this report, we present our work on classification of facial images to predict the gender and age of the person in the image. The methods that we have tried include 1) PCA+SVM, 2) KNN, and 3) Adaboost.

The report is organized as follows. In section 2, we review some of the literature in this subject. In section 3, we discuss our implementation methods in detail. Finally, in section 4, we present the results of our work.

2. Literature review

O'Toole et al [1] describe PCA techniques for gender classification. The paper also describes the structural differences between male and female faces. Wiscott et al [2] present object recognition through graphical representation of facial images, wherein a graph matching technique is used to compare new faces with a set of precomputed graphs, aiding in gender classification.

Y.H. Kwon [3] discusses in detail how facial ratios and wrinkles can be used for successful age classification of images into those of babies, young adults and seniors. Jo Chang-yeon in his report [4], explains the extraction of Local Binary Patterns (LBP) as features for use with a Gentle Adaboost classifier.

Our work is based on a general understanding that we obtained from background research. We were constrained by time, and hence were bound to implement something quick and effective, rather than something that is elaborate and accurate.

3. Our implementation

Our learning problem consisted of 488 color train images, consisting of exactly 4 images each of 122 distinct people. Images spanned both genders and a wide range of ages. Since precise age classification is tough, the problem was modified to classifying age into three groups – below 30, 30-55 and above 55. Our algorithm was tested on a foreign set of 120 color images, consisting of exactly 4 images each of 30 people (not necessarily the same as those on the train images). Both the train and test images were sized 250×250, and the faces were centered, thus essentially eliminating the problem of face recognition.

We took a three step approach to solving this problem. First, we looked at various noise reduction techniques. Second, we focussed on extracting meaningful features for use in learning algorithms. And third, we implemented three different learning algorithms – PCA+SVM, KNN and Adaboost. We elaborate on these in this order.

3.1 Noise reduction

Noise reduction is required for the following reasons. 1) Facial images often come along with a lot of background information that acts as noise. 2) Not all images are obtained under the same illumination and contrast. Faces might themselves be noisy. 3) Although all faces are assumed centered, two facial images do not exactly align with each other in general.

The problem of noisy background is very difficult to solve exactly and requires that the face in the foreground be segmented from the face in the background. Since this is not always easy, we adopted a simpler approach. Since, we have assumed that all faces are centered, we can construct a bounding region centered on the image and consider only the image contents within this region. We experimented with various shapes for the region but the best ones were a facial rectangle, and an elliptical region. From this point onwards, any reference to the image implies a reference to this bounding region containing the face. Figure 1 shows these regions on an image of Lisa Marie Presley.

The problem of different illuminations and contrast

Figure 1: (L 2 R) a. Original image b. Rectangular bounding region, c. Elliptical bounding region

is partly solved by histogram equalization. This involves extending the image histogram to the entire range of pixel values, such that the cumulative distribution of the histogram becomes linear. For color images in RGB space, we need to first convert the image into HSV space and construct the histogram equalized version of *Value* matrix. And then convert the image back to RGB. Figure 2 shows the histogram equalized version of Salma Hayek's facial image.

Additional facial noise called as salt and pepper

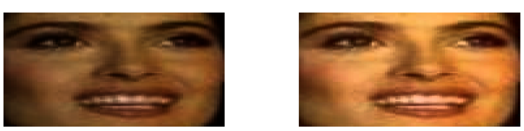

Figure 2: (L 2 R) a. Salma Hayek, b. Effect of histogtam equalization

noise (the presence of unnecessary dots in the image) is eliminated using the median filter. This filter preserves the edges, which we believe are important for recognition. Figure 3 shows the effect of median filtering Carrie Anne Moss's facial image.

For solving the problem of facial alignment, we tried using a face detection algorithm and then scaled the faces to fit the bounding region. Figure 4 shows the effect of scaling the face of Sebastien Grosjean, whose cap occurs within the bounding region. This method is not robust however, because the face detection algorithm is purely based on skin color and suffers from a small false positive and true negative error.

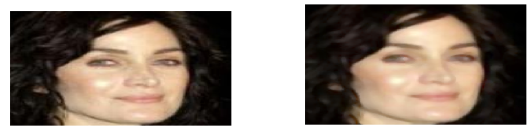

Figure 3: (L 2 R) a. Carrie Anne Moss, b. Effect of median filtering

Figure 4: (L 2 R) a. Sebastien Grosjean, b. Face detected, c. Face scaled

3.2 Features

Extracting meaningful features from the given image data is central to good learning. Features that encode only local pixel information are not informative enough about the overall structure of the image. But local features are easy to compute and extract good statistical performance from. In our work, we have experimented with a wide variety of local feature representations. We describe them here.

A simple feature representation was obtained by concatenating pixel values of the image into a vector. Applying histogram equalization and median filtering before extracting pixel values helped reduce variance across images. Also, before concatenating the pixel values, the image was subsampled to a 25×25 size to further reduce variance. But even with these filters, we recognized that there was significant variance across images of the same person. Thus, we used Principal Components Analysis (PCA) [5] to reduce the dimensionality of the feature and retain only the essential components of the feature vector. Figure 5 shows some eigenfaces obtained by considering only a subset of the basis returned by PCA. We can see that using over a 80 basis vectors does not add much information.

We also experimented with the Histogram of Gradients (HOG) feature [6]. The HOG features of a grayscale image are computed by first running edge detection, and then computing the histogram of edge orientations in cells of 6×6 pixels, folowed by a block normalization procedure. Our primary motivation

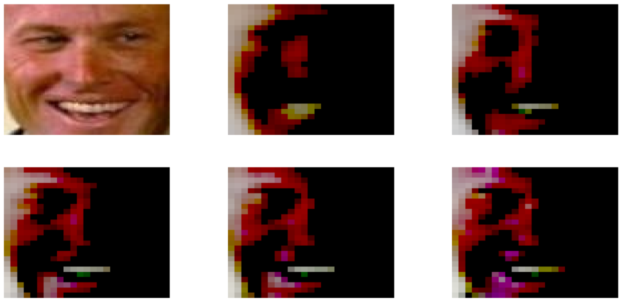

Figure 5: (T 2 B, L 2 R) a. Lance Armstrong, Eigenfaces with b. 20 dims, c. 50 dims, d. 80 dims, e. 100 dims, f. 150 dims

in using HOG features was their illumination invariance and capability to reveal local structure of the image. Figure 6 shows a grayscale image of Catherine Deneuve, and a simplified HOG representation of her image. The pixels in the HOG representation are equal to the average value of edge orientation in the corresponding cell. High pixel values indicate orientation close to 180^o and low pixel values indicate orientation close to 0^o. On careful observation, we can see that the basic structure of the face is retained by these features.

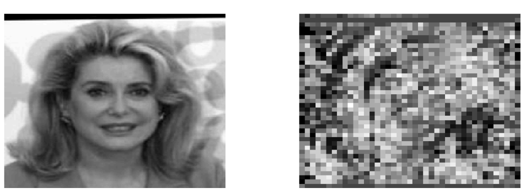

Figure 6: (L 2 R) a. Catherine Deneuve (grayscale), b. Simplified HOG representation

Finally, our third approach was to use Harris Corners to find the important locations in an image, and extract local information from around those pixels. One way to to extract local information from these corners is to consider the local image around them. Another is to compute some form of a histogram. We experimented with Scale Invariance Feature Transform (SIFT) descriptors at these locations. SIFT descriptors are useful because they are invariant to the local scale of the image. Thus, images of the same person with different zoom will produce the same SIFT feature vector. However the one primary difficulty with using Harris Corners in machine learning, in general is that the corners from two images might not be aligned. Thus, for example, the first corner in some image might correspond to the left eyebrow of the person, whereas the first corner in some other image might correspond to the right cheek. This problem can partly resolved by indexing the corners left to right and top to bottom.

3.3 Learning Algorithms

We implemented three learning algorithms. 1) PCA+SVM, 2) KNN and 3) AdaBoost. PCA is an unsupervised learning method that helps reduce dimensionality of a feature vector, while still maintaining as much information as possible. The feature vectors for each image were obtained by simply concatenating the pixel values in the bounding region. The features were kernelized using the Radial Basis Kernel, because it can shatter any finite number of points. We reduced the image into the PCA feature space and used SVM to obtain the separating hyperplanes.

Our second algorithm, KNN is an instance based method that performs classification by finding the K nearest neighbors according to some distance metric and by using either an averaging scheme or majority voting. We used the Chi-squared distance metric defined as follows.

$$\chi^2_{A,B} = \sum_{i=1}^{n} \frac{(A_i - B_i)^2}{A_i + B_i}$$

Here A and B are image feature vectors of two images respectively, and the sum is over all components of these vectors. Since with any local learning method, the features are of central importance, we did not use the entire array of pixel values for KNN. Instead, we ran a Harris Corner detection algorithm and obtained the interest points for every image. And then obtained SIFT desriptors for these corners. Figure 7 shows a set of images with corners (indicated by circles). The 3 nearest neighbors of th leftmost image are shown in each row.

Our third algorithm, adaboost is a discriminative learning classifier that uses a pool of weak (presumably inaccurate) learners and builds up a strong learner. The performance of adaboost is

Figure 7: (L 2 R) Showing 3 Nearest nieghbors based on χ^2 metric on corners

dependent on the choice of weak learners. We used a simple decision stump as the weak classifier. Since thresholding on simple pixel values is not expected to be useful as a weak classifier, we experimented using HOG features with this algorithm.

4. Experimentation results

The experimentation consisted of the same series of steps with all the above algorithms. First, we constructed additional train images, consisting of mirrors of the given images. These were used so as to incorporate into our learning algorithm, the fact that the age or gender of a person is independent of which direction the person is facing in the image. We did consider the option of creating moderately translated and rotated images, but realized that would lead to over fitting the train set.

Next, we applied the various noise reduction techniques to all these train images and extracted the features. Finally, we ran the learning algorithm and used the learned model to make predictions on an unforseen test set. Both the train set and test set were obtained by a partition of the given set of images. We ensured that during partition, the train and test images have no person in common. We expected this to ensure that our learning algorithm was not learning the characteristics of specific people.

Of all our techniques, the PCA+SVM algorithm performed the best. We got age accuracy of 51.4%, and gender accuracy of 81.1% on the foreign test set. These numbers were much better on the given dataset, however. The graphs for PCA and KNN show the learning curves on the given dataset. The x-axis stands for the number of persons (4 images each) used for training. The rest of the images in the given dataset are used for testing. For KNN, the plots correspond to K=5, because we found that gives the best accuracy. The graphs for adaboost were computed across number of boosting iterations. Here the number of train images was chosen such that train/test proportion = given dataset size/foreign dataset size.

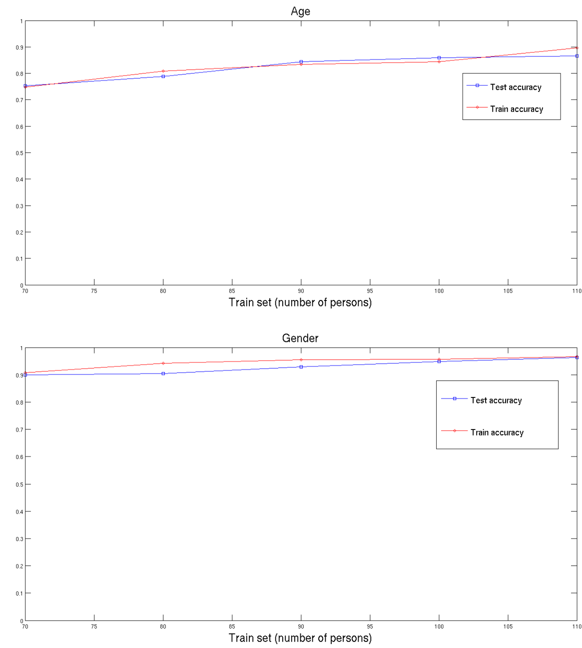

Figure 8: PCA + SVM: Age and gender accuracies for train and test images

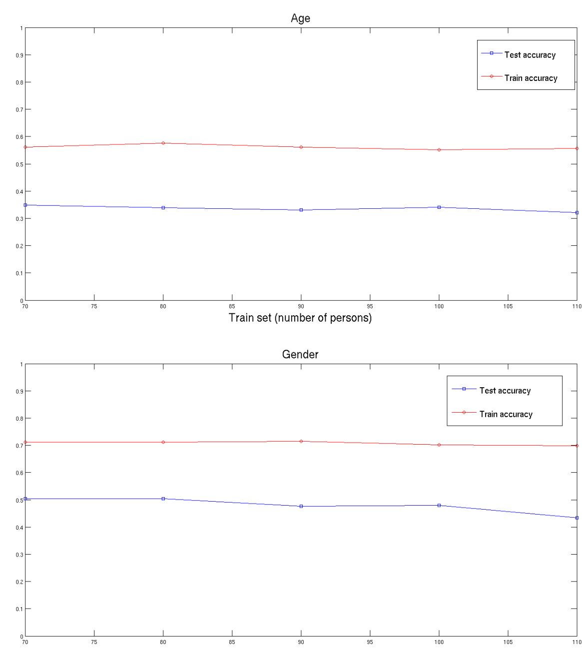

Figure 9: Harris corners + KNN: Age and gender accuracies for train and test images

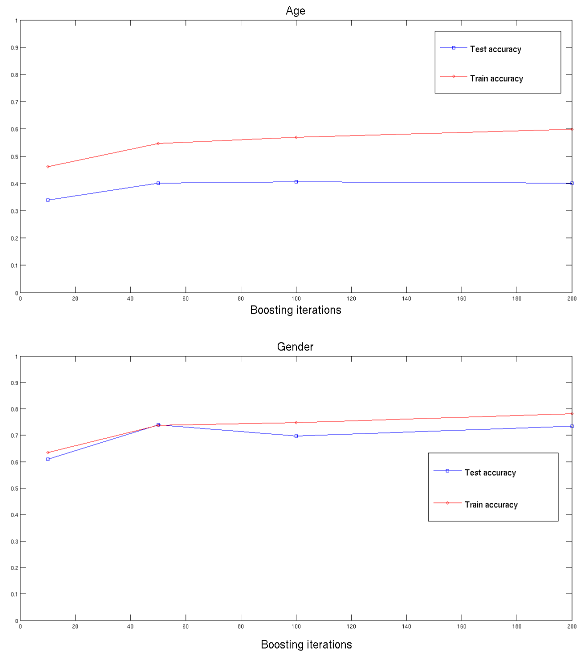

Figure 10: HOG + adaboost: Age and gender accuracies for train and test images

5. Conclusions

Dense feature representations such as concatenated pixel values amend themselves naturally to dimensionality reduction using techniques such as PCA. Of all the algorithms we tried, PCA combined with a radial basis SVM proved to be the best for gender and age classification.

We have used local feature representations. However, improved feature representations that reveal global structure of a facial image would be useful. With such rich set of features, we can expect to see better performance with KNN and adaboost. Also, improved results with Harris corners are expected to be observed by running RANSAC, and estimating the best corner matches between two images.

Acknowledgements

We are grateful to Prof. Ben Taskar, and the TA's for their continued help and guidance throughout our work.

References

[1] OToole J.Alice; Deffenbacher A.Kenneth; Valentin, Dominique; McKee, Karen, Huff, David; Abdi, Herve (1997, Vol.25). The perception of Face Gender: The Role of Stimulus Structure in Recognition and Classification.

[2] Wiskott, Laurenz; Fellous, Jean-Marc; Kruger, Norbert; Malsburg, Christoph (1995). Face Recognition and Gender Determination.

[3] Young Ho Kwon, daVitoria Lobo N (1994). Age classification from facial images.

[4] Chang-yeon, Jo (Dec, 2008). Face Detection using LBP features. CS 229 Final Project Report.

[5] Shlens, Jonathon (Apr, 2009). A Tutorial on Principal Component Analysis.

[6] MATLAB central file exchange 28689-hog-descriptor-for-matlab