This is the published version:


Available from Deakin Research Online:

http://hdl.handle.net/10536/DRO/DU:30006161

Reproduced with the kind permission of the copyright owner.

Copyright : 2006, IEEE
Abstract—A facial classification system that utilises images of faceparts is presented in this paper. Each facepart region is allocated a degree of importance. The random forests approach is employed for classification. The approach grows many classification trees where each tree gives a classification decision. The forest selects the classification that gives the most votes. Experimental results are presented and discussed.

I. INTRODUCTION

Face recognition is an important function of the human visual system and is essential to our social behaviour. Understanding how faces are recognised by the human visual system has been an active area of research in numerous fields of disciplines over the past couple of decades [6,10]. Inspired by physiological clues, a few types of recognition models have emerged. One, which is based on the understanding that the human visual system, focuses on the eyes and mouth areas of a face for recognition. This characteristic of the human visual system can be imitated by a facial classification system by way of assigning different degrees of importance to different parts of the input face image.

Ensemble learning [1] which combines the decisions of multiple classifiers to from an integrated output has emerged as an effective classification method. The variety of the members of an ensemble is known to be an important factor in specifying its generalisation capability. Using ensemble learning, a complex problem can be decomposed into multiple subproblems that are easier to solve. In parallel ensemble classifiers, all individual classifiers are invoked independently, and their results are combined with a combination rule, or a metaclassifier. In cascading ensemble classifiers, classifiers are invoked in a sequential or tree-structures fashion. Inaccurate but fast methods are called upon first, and computationally more expensive but accurate classifiers are left for the later stages.

Random forests [8] is an ensemble learning method that grows many classification trees. To classify an object from an input vector, the input vector is put down each of the trees in the forest. Each tree gives a classification, i.e., the tree votes for that class. The forest selects the classification that has most votes. A feature of random forests is that it does not overfit. It is also fast.

This paper presents a face classification system that first assigns a different degree of importance to each part of a face image. Then it employs the random forests method to classify images.

The paper is organised as follows. Section II briefly describes the ensemble learning approaches. In Section III, the random forests algorithm is explained. Section IV presents and discusses the experimental results. Finally, the concluding remarks are given in Section V.

II. ENSEMBLE LEARNING

Ensemble learning [1] refers to the algorithms that produce collections or ensembles of classifiers which learn to classify by training individual learners and fusing their predictions. Growing an ensemble of trees and getting them vote for the most popular class has provided a good enhancement in the accuracy of classification. Often, random vectors are built that control the growth of each tree in the ensemble. The ensemble learning methods can be divided into two main groups: bagging and boosting.

In bagging, models are fit in parallel where successive trees do not depend on previous trees. Each tree is independently built using bootstrap sample of the dataset. A majority vote determines prediction.

In boosting, models are fit sequentially where successive trees assign additional weight to those observations poorly predicted by previous model. A weighted vote specifies prediction.

III. RANDOM FORESTS

Random forests [2] adds an additional degree of randomness to bagging. Although each tree is constructed using a different bootstrap sample of the dataset, the method by which the classification trees are built is improved. Whilst a node is split using the best split among all variables in standard trees, the node is split using the best among a subset of predictors randomly chosen at that node in a random forest.

A summary of the random forests algorithm for classification is given below [5]:

- **Draw** $n_{tree}$ bootstrap samples from the original data.
- For each of the bootstrap samples, grow an unpruned classification tree, with the following modification: at each node, rather than choosing the best split among all predictors, randomly sample $m_{try}$ of the predictors and choose the best split from among those variables. Bagging can be thought of as the special case of random forests obtained when $m_{try} = p$, the number of predictors.
- Predict new data by aggregating the predictions of the $n_{tree}$ trees, i.e., majority votes for classification, average for regression.

The generalisation error of a forest of tree classifiers depends on the strength of the individual trees in the forest and the correlation between them. Using a random selection
of features to split each node yields error rates that compare favorably to Adaboost, and are more robust with respect to noise. An estimate of the error rate can be obtained, based on the training data, by the following [5]:

- At each bootstrap iteration, predict the data that is not in the bootstrap sample, called "out-of-bag" data, using the tree which is grown with the bootstrap sample.
- Aggregate the out-of-bag predictions. On the average, each data point would be out-of-bag around 36% of the times, so aggregate these predictions. Calculate the error rate, and call it the "out-of-bag" estimate of error rate.

Random forests perform well compared to several other popular classifiers, including discriminant analysis, support vector machines, and neural networks, and is robust against overfitting. In addition, it is user-friendly as it has only two parameters: (i) the number of variables in the random subset at each node, and (ii) the number of trees in the forest. Random forests is not usually very sensitive to the values of these parameters.

Some of the advantages of random forests are listed in the following [8]: (i) for many data sets, it produces a highly accurate classifier; (ii) it handles a very large number of input variables; (iii) it predicts the importance of variables in determining classification; (iv) it generates an internal unbiased estimate of the generalisation error as the forest building progresses; (v) it provides an experimental way to detect variable interactions; (vi) it learns fast; etc.

IV. RESULTS

In this section, the results that are obtained for experiments using the random forests method and also the support vector machine approach [4] are described.

There are a number of face databases that could be used to test the performance of the system. The extended Yale Face Database B is a comprehensive face database that contains many images of several human subjects under different pose and illumination conditions.

The images in the database were captured using an illumination rig. This rig was fitted with 64 computer controlled strobes. The 64 images of a subject in a particular pose were acquired at camera frame rate (30 fps) in about 2 seconds, so there is only small change in head pose and facial expression for images. For each subject, images were captured under 9 different poses. The acquired images are 8-bit grayscale and stored in PGM raw format. The size of each image is 640 × 480.

There exists a cropped version of the extended Yale Face Database B [7] that include front-view images of 38 subjects taken under different illumination conditions. We have used this database in our experiments. The images within the cropped extended Yale Face Database B are manually aligned, cropped, and then resized to 168×192. Some of the image files in this database were found to be corrupt. Therefore, we have used 2414 images from this database.

Figure 1 illustrates sample front-view front-lit face images of the subjects from the cropped extended Yale Face Database B.

In our experiment with random forests, we employed Ting Wang's interface [7] to the random forests algorithm that is developed by Leo Breiman and Adele Cutler [3]. In addition, in our experiments with the support vector machines, we utilised Rong Yan's MatlabArsenal package [9] that encapsulates a number of popular classification algorithms.

A. Experiment 1

In the first experiment, the information from the entire face area was used to train and test the random forests and the support vector machine approaches. All 2414 images from the cropped extended Yale Face Database B were employed in this experiment. The images were grouped into
38 classes. The number of images in each class is as follows: (1,64) (2,64) (3,64) (4,64) (5,64) (6,64) (7,64) (8,64) (9,64) (10,64) (11,60) (12,60) (13,60) (14,60) (15,62) (16,63) (17,63) (18,64) (19,64) (20,64) (21,64) (22,64) (23,64) (24,64) (25,64) (26,64) (27,64) (28,64) (29,64) (30,64) (31,64) (32,64) (33,64) (34,64) (35,64) (36,64) (37,64) (38,64). All images were resized to 56×64. Therefore, the feature number is 3584.

Two datasets were created: train and test. 50% of the images of each class were used to form the train dataset, and the other 50% of the images were used to form the test dataset. Using the random forests as well as the support vector machine classifiers, a number of tests were performed.

With regard to the random forests classifier, we explored: (i) different number of trees to grow, and (ii) different number of variables randomly sampled as candidates at each split. Concerning the support vector machine classifier, we used the support vector machine with the RBF kernel.

Confusion matrices were first calculated for each test. Then classification errors for each class were worked out. Finally, the overall classification error for each test was found. In the following, the classification performances of the random forests as well as support vector machine classifiers on the test dataset are given.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Parameters</th>
<th>Error %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Forests</td>
<td>10 trees 3 variables</td>
<td>38.19</td>
</tr>
<tr>
<td>Support Vector Machines</td>
<td>RBF kernel reduced to 100 dimension</td>
<td>13.96</td>
</tr>
<tr>
<td>Random Forests</td>
<td>100 trees 3 variables</td>
<td>9.44</td>
</tr>
<tr>
<td>Random Forests</td>
<td>100 trees 1000 variables</td>
<td>3.89</td>
</tr>
<tr>
<td>Random Forests</td>
<td>500 trees 500 variables</td>
<td>2.89</td>
</tr>
<tr>
<td>Random Forests</td>
<td>500 trees 100 variables</td>
<td>2.32</td>
</tr>
<tr>
<td>Random Forests</td>
<td>100 trees 100 variables</td>
<td>2.07</td>
</tr>
</tbody>
</table>

B. Experiment 2

In this experiment, each face image is divided into three facepart regions: eyes, nose, and mouth. The size of the extracted eyes, nose, and mouth regions is initially set to 168×62, 168×53, and 168×77, respectively. Then each extracted facepart image is resized according to its degree of importance as follows: eyes to 74×28, nose to 14×5, and mouth to 50×24. These sizes were determined experimentally to achieve the best classification results. Finally, a single feature vector is created out of the three resized facepart images. Therefore, the feature number became 3342.

Figure 3 illustrates sample facepart images of a subject from the cropped extended Yale Face Database B.

All 2414 images from the cropped extended Yale Face Database B were used in the experiment. Two datasets were created: train and test. 50% of the images of each class were used to form the train dataset, and the other 50% of the images were used to form the test dataset. Using the random forests as well as the support vector machine classifiers, a number of tests were performed. The overall classification error for each test was calculated. In Table II, the classification performances of the random forests as well as support vector machine classifiers on the test dataset are provided.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Parameters</th>
<th>Error %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Forests</td>
<td>10 trees 3 variables</td>
<td>30.24</td>
</tr>
<tr>
<td>Support Vector Machines</td>
<td>RBF kernel reduced to 100 dimension</td>
<td>14.43</td>
</tr>
<tr>
<td>Random Forests</td>
<td>100 trees 3 variables</td>
<td>6.79</td>
</tr>
<tr>
<td>Random Forests</td>
<td>100 trees 1000 variables</td>
<td>3.39</td>
</tr>
<tr>
<td>Random Forests</td>
<td>500 trees 500 variables</td>
<td>2.65</td>
</tr>
<tr>
<td>Random Forests</td>
<td>500 trees 100 variables</td>
<td>1.82</td>
</tr>
</tbody>
</table>

C. Experiment 3

In this experiment, we employed the facepart database which is used in the second experiment. However, in this experiment, 90% of the images of each class were used to form the train dataset, and the other 10% of the images were used to form the test dataset. Therefore, the train dataset contained 2173 data and test dataset contained 241 data, each including 3342 features. Using the random forests classifier, a number of tests were performed. The overall classification error for each test was computed. The classification performances are given below.
TABLE III
EXPERIMENT 3: CLASSIFICATION PERFORMANCES

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Parameters</th>
<th>Error %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Forests 10 trees</td>
<td>3 variables</td>
<td>22.40</td>
</tr>
<tr>
<td>Random Forests 100 trees</td>
<td>3 variables</td>
<td>2.07</td>
</tr>
<tr>
<td>Random Forests 100 trees</td>
<td>1000 variables</td>
<td>2.07</td>
</tr>
<tr>
<td>Random Forests 100 trees</td>
<td>100 variables</td>
<td>0.82</td>
</tr>
<tr>
<td>Random Forests 500 trees</td>
<td>100 variables</td>
<td>0.82</td>
</tr>
<tr>
<td>Random Forests 500 trees</td>
<td>500 variables</td>
<td>0.41</td>
</tr>
</tbody>
</table>

D. Discussions

The results obtained in the first and second experiments demonstrate that the facepart approach performs better than full face method in most tests. The lowest classification error was achieved for the facepart approach using the Random Forests algorithm with 500 trees grown, and 100 variables randomly sampled as candidates at each split. However, the best performance was achieved when more data was used in training of the random forests algorithm for the facepart database in Experiment 3. As shown in Table III, the lowest classification error (0.41%) was achieved for 500 trees grown, and 100 variables randomly sampled as candidates at each split.

V. Conclusions

A system was presented that extracts three facepart regions from the full face image and resize them according to their degrees of importance. The random forests and support vector machine approaches were utilised for classification of the facepart images. Three experiments were conducted. The experimental results demonstrate that the classification based on the facepart images produces better results than full face image. The lowest classification error of 0.41% was achieved using the random forests algorithm with 500 trees grown, and 100 variables sampled as candidates at each split.

Acknowledgment

The support of the Australian Research Council (ARC) under a Discovery Grant is gratefully acknowledged.

References