A mixture of two gender classification experts

Yomna Safaa El-Din*, Mohamed N. Moustafa† and Hani Mahdi*
*Department of Computer and Systems Engineering
Ain Shams University
Cairo, Egypt
Email: {yomna.safaa-eldin, hani.mahdi}@eng.asu.edu.eg
†Department of Computer Science and Engineering
American University in Cairo
New Cairo, Egypt
Email: moustafa@ieee.org, m.moustafa@aucegypt.edu

***Abstract*—This paper presents a novel method for combining the outputs of different gender classification techniques based on facial images. Merging the methods is performed by a committee machine using the Bayesian theorem.**

We implement and compare several well-known individual classifiers on four different datasets, then we experiment the proposed machine, and show that it significantly improves the accuracy of classification compared to individual classifiers. We also include results that address the effect of scale on the performance of classifiers.

***Keywords*-committee machines; Bayes; gender classification**

I. Introduction

Facial analysis has been widely investigated in computer vision, including gender, age and expression classification. In particular, gender discrimination is important for several applications; it can improve the performance systems of face verification [1] and face recognition by using separate models for each gender [2], [3], it can help index and retrieve images [4], and it is useful for training interaction systems that behave differently according to the gender of the user.

The accuracy of individual gender classification methods can be boosted by merging more than one classifier [5]. When these classifiers use different input features extracted from the face; there is a higher probability that their false classifications on a set of images are disjoint, in which case merge is helpful to minimize the final error of the combined classifier.

In this paper we propose a committee machine for merging classification methods based on naive Bayesian theorem, and show, on four different image databases, how this combination improves the performance over the best single constituent classifier by up to more than 4%.

The paper is organized as follows; Section II presents an overview of the previous related work. Section III describes the individual classification methods used and Section IV introduces our proposed method for merging these classifiers. Section V explains the experiments we carried and the databases we used, then the results achieved. In the final section, we conclude our work.

II. Related work

A comparative study was performed in [5] for six state-of-the-art gender classification methods to find out their actual reliability. The authors experimented on FERET database and the WWW. They also experimented by combining gender classifier outputs arithmetically, which lead to increased classification accuracies.

A different statistical approach is the single-kernel conditional density estimation system that was introduced in [6] which classifies human faces according to gender, age, ethnicity and other attributes. The author achieved 77% correct gender recognition on datasets from FERET and other diverse video sources.

Convolutional neural networks were used in [7] for gender, age and race recognition. They tested their algorithm on the FG-NET [8] database and another dataset collected from some videos, and achieved 83.53% recognition accuracy of gender on the FG-NET database.

In [9] Faces were represented using elastic graphs labeled with 2-D Gabor wavelet feature, and the system was trained using LDA to classify faces based on their sex, race and expression. They used the complex amplitude of the 2-D Gabor wavelet transform, a grid is then automatically registered with the face using a variant of the elastic graph matching method. The amplitude of the complex valued Gabor transform coefficients are sampled on the grid and combined into a single vector for classification. The author achieved 92% gender classification accuracy on a live demo dataset containing 182 faces, which is a considerably small dataset. However, we think that the restriction of the values taken in the features vector, to be those sampled at the points of a specified grid, may not be the best vector representing the image for classification, in our approach we use Adaboost to select the best Gabor transform coefficients to be used.

The use of committee machines for face recognition was proposed in [10]; they implemented a static committee machine to combine five face recognition algorithms. In our work we adopt the dynamic version of the committee machines; the mixture of experts, for gender classification.

III. Technical background

Before we explain our proposed merge method in the following sections; we explain here the individual methods we used in the combination, and the features used as their inputs.

Four classification methods were used: Least-square SVM [11], Single-Kernel CDE [6], threshold Adaboost and, finally, Convolutional NN. For Adaboost we tried with different input features; either normalized pixel values, or Haar-features [12], or Gabor-filtered images.

A. Single-Kernel CDE

The single-kernel conditional density estimation system proposed in [6] obtains 39 descriptive parameters including gender, age, ethnicity, pose and expression from images of faces. The author used 24x24 images, and concatenated to its vector the other attributes forming a single vector used for training;

$$X = \left[\frac{X_1}{X_2}\right];$$

where X_1 is the attributes' part of the vector and X_2 is the pixels' part.

The training stage mainly calculates the means of X_1 and X_2; μ_1 and μ_2; and the covariance matrix;

$$\Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}.$$

When testing on a new face, only the X2 part of the vector is known; the unknown part X1 is to be expected as;

$$E(X_1) = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(X_2 - \mu_2);$$

which estimates the missing information.

We implemented and used this method as one of the classifiers used in the merges; we only used it to estimate gender. Our attributes vector contains the gender information and the manually labeled eyes positions.

B. Gabor wavelets

The Gabor wavelet representation allows description of spatial frequency structure in the image while preserving information about spatial relations.

A complex-valued 2D Gabor kernel is a plane wave restricted by a Gaussian envelope function [13]:

$$\Psi(k,x) = \frac{k^2}{\sigma^2} e^{-(\frac{k^2 x^2}{2\sigma^2})}[e^{ikx} - e^{\frac{-\sigma^2}{2}}];$$

where k determines the wavelength and orientation of the kernel in image coordinates; $k(\mu,\nu) = k_\nu e^{i\phi_\nu}$, where μ and ν define the orientation and scale of the Gabor kernels, $k_\nu = k_{max} ax/f^\nu$ and $\phi_\nu = \pi\mu/8$. k_{max} is the maximum frequency, and f is the spacing factor between kernels in the frequency domain.

We used Gabor wavelets of five different scales, $\nu \in \{0,\ldots,4\}$ and eight orientations, $\mu \in \{0,\ldots,7\}$, $\sigma = 2\pi$, $k_{max} = \pi/2$ and $f = \sqrt{2}$.

Input images are convolved with the family of kernels and the magnitudes of the complex-valued filter responses are combined into a training vector.

C. Convolutional neural networks

An implementation of convolutional NN was introduced in [14]; that is for recognition of hand-written digits. We adopt their suggested architecture for the CNN, named LeNet-5, shown in Fig. 1.

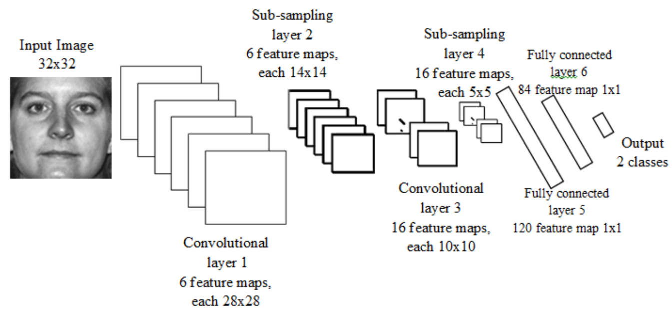

Fig. 1. CNN structure used in the experiments

The general strategy of a convolutional network is to extract simple features at a higher resolution, and then convert them into more complex features at a coarser resolution by sub-sampling.

The LeNet-5 architecture contains seven layers without the input layer. First, two convolutional layers, each followed by a sub-sampling layer; this set can be viewed as a trainable feature extractor.

This set is followed by a trainable classifier to the feature extractor, in the form of 2 fully connected layers (a universal classifier), followed by the output layer.

The first layer receives centered images as input; then for the following layers; each unit in a layer receives inputs from a set of units located in a small neighborhood in the previous layer. The used convolution kernel is of size 5×5. The input faces are sized to 28×28, and then padded to 32×32 for the input layer.

IV. Proposed method

This section explains our proposed committee machine for merging the outputs of more than one classifier for a better final decision.

A. Mixture of experts

It is a form of dynamic committee machines, in which the input signal is directly involved in actuating the mechanism that integrates the outputs of the constituent experts (classifiers) as shown in Fig. 2. The constituent outputs are non-linearly combined by some form of gating system to produce an overall output that is superior to that of any single expert alone.

In our form of committee machine we use naive Bayes at the combiner stage. We build several machines, each combining two experts.

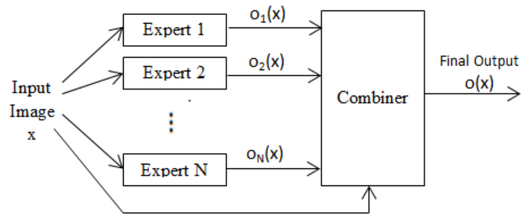

Fig. 2. Structure of a general Mixture of Experts network, where the input x influences the output of the combination, o

B. Naive Bayes probabilistic model

A naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong, yet naive, independence assumptions. The Bayes' theorem;

$$p(C|F_1,\dots,F_n) = \frac{p(C)p(F_1,\dots,F_n|C)}{p(F_1,\dots,F_n)}$$

can be understood as;

$$posterior = (prior \times likelihood)/evidence$$

In practice we are only interested in the numerator of that fraction, since the denominator does not depend on the class C and the values of the features F_i are given.

Given a new features-vector, the classifier then chooses the class which has the maximum posterior.

C. Proposed Mixture of experts with Bayesian combiner

We apply this Bayes' theorem in our committee machine as explained in this section.

1) Training: For each individual classification method, we train it on a training subset of the database images, for each database alone. We then train the Bayesian merger using another subset of images. Training the merger mainly means calculating the prior of each of the contributing individual methods to be merged and likelihood of their outputs' confidences.

For each individual classification method, we run its previously-trained classifier on this subset and obtain the following:

- *Prior:* The achieved correct rate. At merge time we normalize it with the other method's prior, so that $\sum_{m=1}^{n} prior_m = 1$; where n is the number of methods to be merged.

- *Likelihood:* We use the confidences returned by the classifier from this subset then split the confidence range into N intervals. For each interval n$\in$ $\{0,\dots,N\}$, we calculate the percentage of images that were correctly classified (true percent) and those that were wrong (false percent). Likelihood of this interval is the true percent minus the false percent.

If an interval has zero images in it, then its likelihood is taken to be the average of the two intervals around it; $n-1$ and $n+1$. If an interval has the false percent larger than the true percent, then a likelihood value will be < 0, so at the end, we shift all the values to have a $min = 1$. Likelihood values are then normalized to $[0-1]$, to finally obtain a discrete conditional probability table of N entries. For our approach we chose $N = 20$.

2) Merging: When a new face is to be classified using the merge of more than one method, the trained classifier of each method alone is run on the input image and each returns an output o_m along with its confidence c_m.

The likelihoods of these confidences are retrieved, using the trained Bayesian merger; also the prior of each method alone is retrieved. We normalize the priors by calculating the following:

$$sum\_p = \sum_{m=1}^{n} prior_m$$

And for each method m

$$pr(m) = prior_m/sum\_p$$

$$posterior(m, c_m) = pr(m) \times likelihood(m, c_m)$$

The final output (class) is then taken to be that of the method with the highest posterior.

D. Enhanced mixture of experts

For additional accuracy, we add another expert to the machine, that will be fused to the machine in a different way than the other experts. When using a machine with two experts, we run the trained classifier of each expert on the input image and each method returns a classification output for this image. If they both agree, nothing more is done, but in case each expert decides the image to belong to a different class, only then the additional third expert is to be invoked; before running the Bayesian merge. It this extra expert returns a result with a high confidence value, its decision is taken regardless of the other methods' decisions. However, if the result returned has a low confidence, it is discarded, and the Bayesian merge explained in the previous subsection is performed to choose among the main experts' decisions. This approach increased the accuracies obtained than the Bayesian merge alone.

E. Merge combinations

In order to obtain improved results by merging the outputs of more than one classifier; the individual classifiers merged should be chosen based on their errors pattern. That is on a set of test faces, each method alone mis-classifies a subset of these faces; for maximum benefit of the merge we choose classifiers with minimum intersection in their error subsets.

This minimum overlap is achieved by using classifiers that rely on different cues for their decisions. For example, SVM and Single-Kernel (SK) use image pixels, either raw or their normalized values; while Adaboost uses Haar-features, or Gabor-filtered images which rely on directional features related to edges and ridges. On the other hand, convolutional NN deals with slight rotation, scale, and shift discrepancies and extract texture based features.

V. Experiments and Results

In this section we describe the data we used, along with the type of experiments performed and the results of these experiments.

TABLE I
NUMBER OF FACES USED FOR EACH DATABASE.

Database	Train	TrainExtra	Test
	(50% Male, 50% Female)		(M/F)
FERET	620	800	200/188
MixDB	1000	474	412/359
LFW	1000	800	500/500
KinFace	140	124	50/30

A. Datasets

We used four image databases in the experiments; the FERET image database, the Labeled Faces in the Wild (LFW) [15], the UB KinFace database [16], [17], [18] and, finally, a dataset we refer to as MixDB which contains images of people collected privately including several ethnicities; Caucasians, Asians and also some from African descent. On the other hand, the LFW database offers a unique collection of annotated faces captured from news articles on the web. The UB KinFace database contains 600 images most of which are real-world collections of public figures from Internet. This dataset is mainly used for the purpose of kinship verification; while we present here the first published gender classification results on this dataset. From the FERET database we used the fa- and fb-subsets that contain frontal faces.

Images used in the experiments are manually prepared as follows:

- Colored images are transformed to gray,

- Images are rotated to be horizontal using manually labeled eyes positions,

- The images are then cropped to include only the face; dimensions are a function of the inter-eyes distance.

For each DB, we created three subsets; one for training individual classifiers; another subset for training the merging classifier; we call it TrainExtra subset, and, finally, one for testing.

For training, duplicate images of the same person were removed, so that only one image per person was left, in FERET, and two images per person, in LFW database.

Table I shows the number of faces that we hand-labeled and used for each database.

B. Experiments

To validate our proposed technique, we carried on a series of experiments for which we used MATLAB to implement and test.

First experiment: We run each of the individual methods used in our machine.

- SVM: We used MATLAB's bioinformatics toolbox.

- SK: From [6] details we implemented the algorithm using MATLAB.

- Adaboost: We also used MATLAB for implementation of the feature extraction then training of Adaboost. The number of features we use is equal to the size of the input face; i.e. for 24×24 face image, we used $T = 576$. For Adaboost on Gabor-filtered faces, we used kernels of five different sizes and eight orientations as explained in Section III.

TABLE II
WEIGHTED AVERAGE OF INDIVIDUAL CLASSIFIERS OVER ALL DATASETS

Method	Features	15×15 **faces**	24×24 **faces**
SVM	Raw	85.56%	81.14%
	Normalized	85.78%	80.02%
SK	Raw	83.93%	74.09%
Adaboost	Normalized	83.2%	81.02%
	Haar	85.31%	84.18%
	Gabor	80.57%	80.60%
CNN (28×28)	Raw	-	80.51%
	Normalized	-	81.40%

- CNN: We used the architecture shown in Fig. 1, so we used images of size 28×28.

Second experiment: We run our proposed merge machine on the same data showing its enhancement. We tried several merge combinations which will be explained in the following section.

Both experiments were performed on two sizes each; 15×15 and 24×24. The sizes of faces were reduced using bi-cubic interpolation.

For each size, tests were run two times, once on the input faces as they are, and the other time, on preprocessed faces. For preprocessing we used intensity adjustment, which is an image enhancement technique that maps an image's intensity values to a new range. We used the MATLAB function *imadjust* which increases the contrast in a low-contrast gray-scale image by remapping the data values to fill the entire intensity range $[0 - 255]$. We tried histogram equalization for preprocessing, but on the average it gave worse results, so we excluded it from the results. The shown results are the average of the two runs on the four datasets used.

C. Results of individual classifiers

We include here the comparative results of the individual classifiers we used. SVM was experimented on raw image pixels and on normalized pixel values. Single Kernel (SK) method was run on raw faces; while Adaboost was run on three different features; normalized pixel values, Haar features and Gabor wavelets.

Table II presents the weighted average results for each method on all four datasets, from which we notice that on the average all methods show better results on smaller faces (15×15).

We can also see that for SVM, normalizing the image pixels doesn't show much improvement on the average; while for CNN, the average shows slight improvement using normalized pixels than raw pixels. So when merging classifiers, in the following subsection, we chose to include only SVM on raw pixels, and CNN on normalized pixel values.

D. Results of merged classifiers

In the following figures; Fig. 3 and Fig. 4, we present the results obtained by running our merging approach. The results shown are the weighted average on all four datasets.

We choose the classifiers to merge, based on the intersection of their error patterns; as will be explained later on Table III.

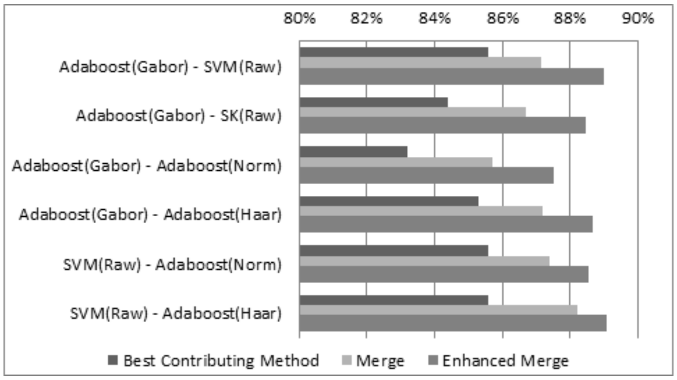

Fig. 3. Merge results on 15×15 faces

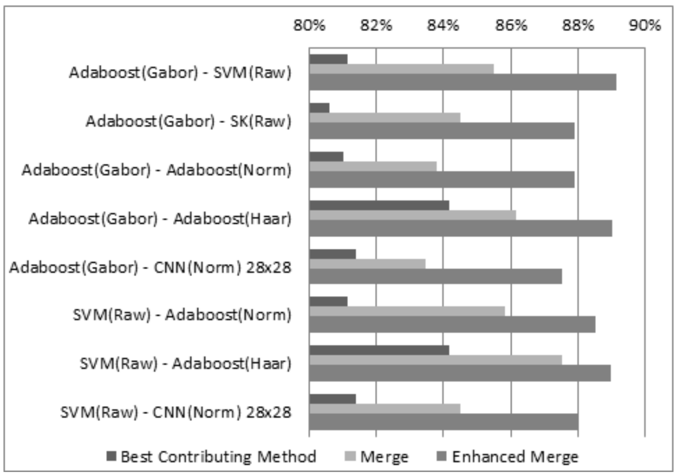

Fig. 4. Merge results on 24×24 faces

Merge1: We merge Adaboost on Gabor-filtered faces with the rest of the classifiers listed in the previous subsection.

Enhanced Merge1: Same as Merge1, but using Neural Networks as an extra expert trained with normalized pixel values of the face images, in the way explained in Subsection IV-D. We accept results of this extra NN expert only if the main two experts disagree, and the NN's returned confidence is above 70%.

Merge2: SVM on raw pixels is merged with the other classifiers except the SK, as they both use the same raw pixel values.

TABLE III
SK (ON RAW IMAGE PIXELS) VS. SVM (ON RAW IMAGE PIXELS) AND ADABOOST (ON GABOR-FILTERED IMAGES)

Method (1)	**Method (2)**	**Size**	**Different decisions %**	**% SK correct**	**% The other method (2) correct**
SK (Raw)	**SVM (Raw)**	15×15	1.91%	0.40%	1.52%
		24×24	11.33%	2.55%	8.78%
	Adaboost (Gabor)	15×15	21.51%	12.77%	8.74%
		24×24	30.60%	12.39%	18.20%

TABLE IV
FERET RESULTS: ENHANCED MERGE2A: SVM (RAW) AS METHOD (1) MERGED WITH ADABOOST ON 15×15 FACES

Method (1) result	**Method (2)**	**Features**	**Method (2) result**	**Merge results**	**% Increase over the best contributing method**
88.42%	**Adaboost**	Normalized	87.89%	**92.37%**	3.95%
		Haar	90.53%	**93.03%**	2.50%
		Gabor	85.13%	**92.63%**	4.21%

TABLE V
FERET DATABASE COMPARISON

		Size	**Best Individual**	**Best Merge**	**% Increase over Best Individual**
Our Work	**Normalized**	15×15	90.53%	93.03%	2.50%
		24×24	92.37%	93.68%	1.32%
Mäkinen [5]	**Normalized**	24×24	92.22%	-	-
	Without Normalization	24×24	84.44%	85.71%	1.27%
		32×40 (With Hair)	90.07%	92.86%	2.79%

Enhanced Merge2: Same as Merge2, but enhanced with NN like Enhanced Merge1.

Results of both merges and their enhanced versions are compared in Fig. 3 and Fig. 4

We justify our choices by running the methods first on our TrainExtra dataset and study the similarities and differences between their outputs. For example, Table III elaborates on the similarity of decisions of SK and SVM on raw image pixels, while when compared to Adaboost (Gabor) they show different behavior. It can be seen from that table, that even when SVM and SK have 11% different decision (@ 24×24 faces); SVM prevails, i.e. SVM gives better results for almost 80% of these different cases.

On the other hand, when comparing SK and Adaboost (Gabor) we can see they have a higher difference percentage; which is well divided between the both of them; in which case using the merge machine is a good idea to benefit from the correct decisions of each classifier alone.

It can be noticed from the included results in Fig. 3 and Fig. 4, that using our proposed mixture of experts leads to better classification than each individual classifier; and the enhanced merge achieves even better accuracy. The increase in performance is more significant on higher resolution faces which showed less correct rate than the 15×15 faces when using individual methods.

E. Detailed results and comparison on FERET database

We like to detail here in Table IV the results we obtained using our proposed mixture of experts machine on FERET dataset, instead of the weighted averages listed above. We include results only using *Enhanced Merge2* on 15×15 images, as it obtains the best merge results.

TABLE VI
INTERNET-IMAGES COMPARISON

		Size	Best Individual	Best Merge	% Increase over Best Individual
Our Work [LFW]	**Normalized**	15×15 24×24	84.70% 87.55%	87.90% 89.15%	3.20% 1.60%
Mäkinen [5] [WWW]	**Without Normalization**	24×24 32×40 (With Hair)	78.28% 76.61%	81.00% 83.14%	2.72% 6.43%

TABLE VII
BEST RESULTS ON 15×15 FACES

	Best Individual	Best Merge	% Increase over Best Individual	Best Enhanced Merge	% Increase over Best Individual
FERET	90.53%	93.03%	2.50%	93.03%	2.50%
LFW	84.70%	86.70%	2.00%	87.90%	3.20%
MixDB	86.96%	88.72%	1.75%	90.53%	3.57%
KinFace	89.38%	88.75%	-0.63%	90.00%	0.62%

The comparison work of [5] tested six gender classification methods on FERET database and another dataset named WWW containing images collected from the web. They applied several settings,

- with vs. without hair: without hair 24×24 faces, while they increased the size to include the hair to be 32×40.

- with vs. without normalization of images based on the manually located eyes position (for FERET only, while for WWW, they detected faces automatically, so they only performed tests on non-normalized faces).

For our work, we adopt only the normalized images, without hair.

They also combined all six classifier outputs together using four types of combination methods; (refer to [5] for more details), and they achieved improvement in accuracy as shown in Tables V and VI.

VI. CONCLUSIONS

In this paper, we proposed a new technique to combine two classifiers in a mixture of experts' committee machine, for which we used a naive Bayes approach for the merge. We also introduced an enhancement of this 2-method machine by adding a third expert used in a specific way. We implemented and compared several state-of-the art gender classification algorithms, on four different benchmark databases, using two resolutions. We validate that using high resolution images doesn't necessarily lead to better classification accuracy, which we showed using SVM and SK classifiers. The best results were obtained using Adaboost on Haar features, which showed more than 92% correct rate on FERET database, 87.55% on LFW dataset and 91.25% on KinFace, all using 24×24 images, while for the MixDB dataset, SVM obtained best performance of 86.96% on 15×15 face images.

We also evaluated our proposed method on these databases, and showed an improvement in the performance over the best classifier working on its own. On the average (the weighted average of results on all four datasets), best results obtained using individual classifiers are for Adaboost on Haar-features and SVM using 15×15 images, which are above 85%. For this case we obtained an improved accuracy of 89.31% by merging them.

Using our proposed committee machine, we obtained best results shown in Tables VII and VIII for each database separately. Most of the results in these tables are obtained using *Enhanced Merge2*; that is merging SVM with Adaboost, and NN as an extra expert; while some were obtained by merging Adaboost on Gabor-features with Adaboost on Haar-features, also with NN as an extra expert. Best validation of the effectiveness of our proposal in enhancing the accuracies of its contributing experts is shown using *Enhanced Merge* on 24×24 images, where more than 7% improvement is achieved.

In this work we focused on merging two classifiers only; for future work we would like to see if merging more than two different classifiers would be more efficient.

TABLE VIII
BEST RESULTS ON 24×24 FACES

	Best Individual	Best Merge	% Increase over Best Individual	Best Enhanced Merge	% Increase over Best Individual
FERET	92.37%	90.92%	-1.45%	93.68%	1.32%
LFW	87.55%	89.15%	1.60%	88.95%	1.40%
MixDB	85.02%	87.03%	2.01%	90.73%	5.71%
KinFace	91.25%	91.88%	0.62%	92.50%	1.25%

REFERENCES

[1] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in *In IEEE International Conference on Computer Vision (ICCV)*, 2009.

[2] G. Givens, J. R. Beveridge, B. A. Draper, P. Grother, and P. J. Phillips, "How features of the human face affect recognition: A statistical comparison of three face recognition algorithms," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 2, pp. 381–388, 2004.

[3] J. R. Beveridge, G. H. Givens, P. J. Phillips, and B. A. Draper, "Factors that influence algorithm performance in the face recognition grand challenge," *Computer Vision and Image Understanding*, pp. 750–762, 2009.

[4] N. Kumar, P. N. Belhumeur, and S. K. Nayar, "FaceTracer: A Search Engine for Large Collections of Images with Faces," in *European Conference on Computer Vision (ECCV)*, Oct 2008, pp. 340–353.

[5] E. Mkinen and R. Raisamo, "An experimental comparison of gender classification methods," *Pattern Recognition Letters*, vol. 29, pp. 1544–1556, July 2008.

[6] J. A. Robinson, "Regularized single-kernel conditional density estimation for face description," in *Proceedings of the 16th IEEE international conference on Image processing*, ser. ICIP'09, 2009, pp. 2433–2436.

[7] M. Yang, S. Zhu, F. Lv, and K. Yu, "Correspondence driven adaptation for human profile recognition." in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR*, June 2011, pp. 505–512.

[8] FG-NET aging database. [Online]. Available: http://www.fgnet.rsunit.com/

[9] M. J. Lyons, J. Budynek, A. Plante, and S. Akamatsu, "Classifying facial attributes using a 2-d gaborwavelet representation and discriminant analysis," in *Proc. 4th International Conference on Automatic Face and Gesture Recognition*, Grenoble, France, 28-30 March 2000, pp. 202–207.

[10] H.-M. Tang, M. Lyu, and I. King, "Face recognition committee machine," *Multimedia and Expo, IEEE International Conference on*, vol. 3, pp. 425–428, 2003.

[11] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Process. Lett.*, vol. 9, no. 3, pp. 293–300, Jun. 1999.

[12] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), Vol. 1*, 2001, p. 511518.

[13] M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lande, C. von der Malsburg, R. P. Wurtz, and W. Konen, "Distortion invariant object recognition in the dynamic link architecture," *Computers, IEEE Trans.*, vol. 42, pp. 300–311, August 2002.

[14] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, 1998, pp. 2278–2324.

[15] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07-49, October 2007.

[16] S. Xia, M. Shao, J. Luo, and Y. Fu, "Understanding kin relationships in a photo," *IEEE Transactions on Multimedia*, 2012.

[17] S. Xia, M. Shao, and Y. Fu, "Kinship verification through transfer learning," in *Proc. International Joint Conferences on Artificial Intelligence IJCAI*, 2011, pp. 2539–2544.

[18] M. Shao, S. Xia, and Y. Fu, "Genealogical face recognition based on ub kinface database," in *Proc. IEEE CVPR Workshop on Biometrics BIOM*, 2011.