Comparison of Several Classification Algorithms for Gender Recognition from Face Images

Mutlu Sakarkaya, Fahrettin Yanbol and Zeyneb Kurt
Computer Engineering Department, Yildiz Technical University, Davutpasa Campus,34210 Esenler,Istanbul, Turkey
{ l6205501, l1106037}@ std.yildiz.edu.tr, zeyneb@ce.yildiz.edu.tr

Abstract—**This paper presents a comparison between several algorithms which were employed for gender recognition automatically. Firstly, the face images of various mature women and men samples were gathered, and face images were separated as train dataset and test dataset. Both of the datasets were pre-processed and made ready for following operations. Secondly, Principal Component Analysis (PCA) was applied to train dataset to extract the most distinguishing features. Finally, three classification algorithms, Support Vector Machine (SVM), k-Nearest Neighbourhood (k-NN), and Multivariate Classification with Multivariate Gauss Distribution (MCMGD) algorithms were implemented and compared to determine the most suitable and successful algorithm for gender recognition from face images. Experimental results illustrate that k-NN with k values 5, 7, 9 outperformed the other approaches.**

I. INTRODUCTION

In the proposed study, gender recognition by using face images was aimed via using several classification algorithms. Many studies exist for gender recognition by using face images. The first applications had been presented by Cottrel and Metcalfe [1] and Golomb et al. [2] concurrently. PCA and an Artificial Neural Network had been used for gender recognition from face images [3]. A neural network application with very low resolution (8x8) face images had been proposed in [4]. A study for face and gender recognition via using Elastic Bunch Graph Matching algorithm had been presented in [5]. Gabor wavelets with PCA and Linear Discriminant Analysis (LDA) had been used in [6]. A threshold Adaboost classifier had been used for gender recognition [7]. A feature selection system via using Genetic Algorithm for gender recognition was presented in [8]. A gender recognition system by shapes obtained from shading and weighted principal geodesic analysis had been developed in [9]. Makinen and Raisamo achieved an experimental comparison of gender classification methods [10]. We compared several algorithms for gender recognition, too; unlike their study we also searched the effect of the feature number on the performance of the algorithms.

Furthermore, in the most commonly proposed face recognition applications, SVM [11-13], 3D Morphable Models [14], Texture Representation [15] algorithms, and Statistical Models of Appearance [16] had been used.

In the proposed system, we have compared three different learning approaches to determine the best one for gender recognition from face images. Furthermore the effect of extracted feature numbers on the algorithms' performance was also analyzed. The face images were collected from various places, thus they did not have a standard and common size. Therefore, the face images of men and women should have been pre-processed before learning and testing steps. Besides, there were a wide range of features in the system. Thus, dimension should have been reduced before training and testing the algorithms. However while the dimension is reduced, the most prominent features should be protected. PCA was used for efficient feature extraction. After the feature extraction of train dataset, the first learning algorithm, SVM, was trained. In the testing phase of SVM, images in the test dataset were also pre-processed and became ready for testing. Features of test dataset were also extracted with respect to eigen vectors of train dataset. Learned SVM structure classified the test dataset.

After feature extraction of both datasets, the second algorithm, k-NN, classified a test sample according to k closest Euclidean Distance between all images in the train data and the given test image.

In the last learning algorithm, MCMGD, we assumed that class-conditional densities have normal distribution. After feature extraction of both datasets, Gaussian parameters of male and female classes were calculated by using train dataset. Test images were evaluated according to the parameters of those Gauss Distributions. The biggest Gauss Distribution membership value had indicated the class label of the given test image.

The outline of the paper is: Section 2 describes pre-processing steps of the system. Section 3 presents extraction of the most distinctive features of the images by using PCA. In section 4, learning algorithms which were used in the system are described. In section 5, experimental results are given for comparison of those three algorithms. Finally in the last section conclusion is given.

II. PREPROCESSING

The face images of men and women were obtained from various places, such as internet, several image repositories, students and employees in our university. Because the locations and scales of the face regions in the images were not the same, firstly images in the train and test datasets should be pre-processed.

All the images are gray level in our datasets. Firstly, the potential noises were eliminated by an averaging filter. Then, histogram equalization was used to increase contrast. After that we needed to overlap the face regions of all images. To achieve this, the coordinates of the midpoints between two eyes were fixed to the same position and the distance between two eyes was also fixed, for all the images. Therefore face images were aligned,

978-1-4673-2695-7/12/$31.00 ©2012 IEEE

resized, and overlapped. After all, a mask was applied to images in order to eliminate unnecessary areas of these images.

The result of the pre-processing steps does not only make face images standard, but also it makes ready these images for feature extraction. The results of each step in pre-processing operation are given in the Figure 1. Finally all images in training and test datasets become 60x50 pixel. All of the pre-processing steps were achieved automatically, not manual.

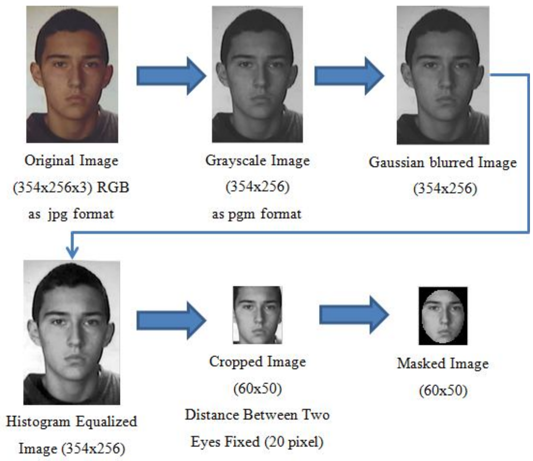

Figure 1. Pre-processing steps of proposed system.

III. Feature Extraction by PCA

Turk and Pentland applied PCA in face recognition for extracting the characteristic features of faces [17]. In this study PCA was used to reduce dimension of original data while retaining most of the variation in the data set.

PCA's procedure can be thought as revealing the internal structure of the data in a way which beest explains the variance in the data. If a multivariate dataset is visualised as a set of coordinates in a high-dimensional data space, PCA can supply the user with a lower-dimensional picture, a "shadow" of this object when viewed from its most informative viewpoint. This is done by using only the first few principal components so that the dimensionality of the transformed data is reduced [18].

Each image in the training dataset was transformed to a vector. One image had a size of 60x50 pixels and it was transformed to a 3000x1vector. 20 female and 20 male images were used for training. All vectors of 40 images in the training dataset were gathered in a matrix.

The mean vector of the train data was calculated. Mean centered vectors were obtained by subtracting the mean from each vector of the training data. Each mean centered data is collected in a matrix (A) as each column belongs to a person.

The covariance matrix (C) of mean centered matrix (A) is calculated by Eq.1:

$$\mathbf{C} = \mathbf{A} \times \mathbf{A}^{\mathbf{T}} \tag{1}$$

Eigen values ($\boldsymbol{\lambda}$) and vectors (**V**) of C are obtained:

$$\mathbf{C} \times \mathbf{V} = \boldsymbol{\lambda} \times \mathbf{V} \tag{2}$$

Eigen face is obtained for a training sample (**X**) by (3):

$$\mathbf{Y} = \mathbf{V}^{\mathbf{T}} \times \mathbf{X} \tag{3}$$

Test images should be multiplied by the eigen vectors (**V**) of the train dataset. Now, the datasets are ready for training and testing the classification algorithms.

IV. Classification Algorithms

In this section we describe the classification algorithms SVM, k-NN, and MCMGD.

A. *Support Vector Machine (SVM)*

SVM is a supervised learning method that is used for classification. SVM was introduced by Vapnik [19]. SVM constructs a hyper plane which has the largest distance to the nearest training data points of any class, since in general the larger the margin the lower the generalization error of the classifier.

Since SVM is a classifier for two-class classification problem, SVM training algorithm builds a model that predicts whether a new example falls into one class or the other. Each instance in the training set contains one output value. The goal of SVM is to produce a model which predicts target value (class label) of data instances in the testing set successfully.

The decision boundary hyperplane $f(x)$ is defined by (4) where k is kernel function and b is bias value.

$$f(x) = \sum_{i=1}^{M} y_i \alpha_i \, k(x, x_i) + b \tag{4}$$

Linear kernel of SVM was used in this system.

B. *k-Nearest Neighbourhood (k-NN)*

The k-NN algorithm is one of the simplest machine learning algorithms. There is not any learning step of the algorithm, only the feature vectors and class labels of the training samples should be restored. In the classification process a test sample, whose class label is not known, is classified by a majority vote of its neighbours, with the purpose of being assigned to the most commonly seen class in its k nearest neighbours. In this scheme, a test sample is assigned to the class c if this class is the most frequent class label among the k nearest training samples. Especially, in two-class classification problems, k should be chosen as an odd number as this avoids tied votes. In this study k was chosen as {1, 3, 5, 7, 9}. Results are illustrated in section 5.

Distance between the test sample and all samples in the training dataset should be calculated by a particular metric. It is usual to use Euclidean distance. The other metrics such as Manhattan distance could be also used. In this study Euclidean distance was used. It is given in (5):

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_N - y_N)^2} \tag{5}$$

Training and test samples are vectors in a multidimensional feature space. x_i and y_i are the i[th] feature of the sample x and sample y respectively.

C. Multivariate Classification with Multivariate Gauss Distribution (MCMGD)

In the third algorithm we assumed that distribution of the class-conditional densities is Normal Density:

$$p(\mathbf{x}|C_i) = \frac{1}{(2\pi)^{d/2}|\mathbf{\Sigma}_i|^{1/2}} exp\left[\frac{-1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \mathbf{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right] \quad (6)$$

where $\mathbf{x}$: multivariate test sample; $\boldsymbol{\mu}_i$ and $\mathbf{\Sigma}_i$ denote estimation of mean vector and covariance matrix of sample distribution of i[th] class (C_i). $\boldsymbol{\mu}_i$ and $\mathbf{\Sigma}_i$ were calculated for each of two classes, by using training dataset, by (7) and (8):

$$\boldsymbol{\mu}_i = \frac{\sum_{x_j \in C_i} x_j}{Number\_of\_samples\_in\_C_i} \quad (7)$$

$$\mathbf{\Sigma}_i = \frac{\sum_{x_j \in C_i} (x_j - \boldsymbol{\mu}_i)(x_j - \boldsymbol{\mu}_i)^T}{Number\_of\_samples\_in\_C_i} \quad (8)$$

The main reason of this assumption is because of its analytical simplicity. Besides, the normal density is a model for many naturally occurring phenomena. While the real data may not be exactly multivariate normal often, it is a useful approximation. In addition to its mathematical tractability, the model is robust to departures from normality [18]. After that assumption, a test sample can be classified according to posterior density that is given in (9):

$$p(C_i|\mathbf{x}) = \frac{p(x|C_i)p(C_i)}{\sum_{j=1}^{K} p(x|C_j)p(C_j)} \quad (9)$$

where K: is the number of classes; $p(C_i)$:is prior probability of C_i ; $p(C_i|\mathbf{x})$: is the probability of belonging to i[th] class (C_i) for a test sample ($\mathbf{x}$). Equation (10) is obtained by writing (9) in logarithmic form:

$$g_i(\mathbf{x}) = log\ p(x|C_i) + log\ p(C_i) \quad (10)$$

where $g_i(\mathbf{x})$: is the discriminator function of C_i. The denominator of the (9) is the same for all of the classes, thus it is discarded. Discriminator function $g_i(\mathbf{x})$ can be written as in (11) by using the assumption in (6) [18]:

$$g_i(\mathbf{x}) = -\frac{d}{2}log 2\pi - \frac{1}{2}log|\mathbf{\Sigma}_i| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \mathbf{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i) + log\, p(C_i) \quad (11)$$

To sum up the operations in the last algorithm: Firstly $\boldsymbol{\mu}_i$ and $\mathbf{\Sigma}_i$ for each class must be calculated separately by using train dataset as in (7) and (8). After that, also for each class $g_i(\mathbf{x})$ values of a test sample ($\mathbf{x}$) should be calculated via using (11). The biggest $g_i(\mathbf{x})$ value will depict the class membership of the test sample ($\mathbf{x}$).

V. Experimental Results

There are totally 70 images in our datasets. Each class (male and female) has 35 samples. 40 images are used for learning, 30 images are used for test. Number of samples according to classes and datasets is given in Table 1.

TABLE II.
NUMBER OF SAMPLES

	Female	Male	Total
Train set	20	20	40
Test set	15	15	30
Total	35	35	**70**

Size of the images in the datasets is 60x50. Thus, number of feature is 3000. PCA was used for dimension reduction as described in previous chapters. We firstly reduced the dimension to 100. Then we evaluated the success ratios of three learning algorithms. After that, we incremented the dimension by 200. Again the success ratio of the algorithms was evaluated. Until the dimension reached 2900, at each step we incremented the dimension by 200.

The success ratio of SVM linear kernel (for each class separately) according to reduced dimension is shown in Figure 2 and Table 2.

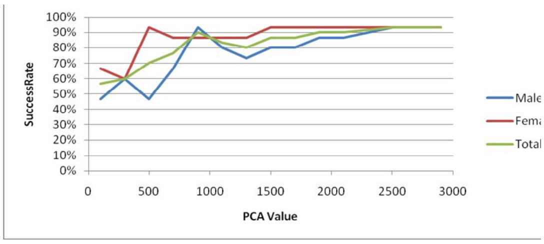

Figure 2. Success ratio of SVM according to dimension.

TABLE I.
SUCCESS RATIO OF SVM ACCORDING TO DIMENSION

Reduced Dimension	Male Class	Female Class	Total Success
100	47%	67%	57%
300	60%	60%	60%
500	47%	93%	70%
700	67%	87%	77%
900	93%	87%	90%
1100	80%	87%	83%
1300	73%	87%	80%
1500	80%	93%	87%
1700	80%	93%	87%
1900	87%	93%	90%
2100	87%	93%	90%
2300	87%	93%	90%
2500	93%	93%	93%
2700	93%	93%	93%
2900	93%	93%	93%

The success ratio of k-NN when *k=1* according to reduced dimension is illustrated in Figure 3. The success ratio of k-NN when *k=1* for each class and through the all dimensions is the same, 80%.

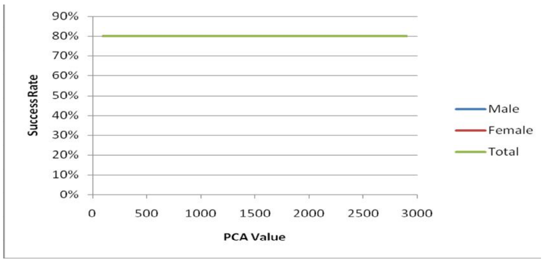

Figure 5. Success ratio of k-NN according to dimension (k=1).

The success ratio of k-NN when *k=3* according to reduced dimension is shown in Figure 4. The success ratio of k-NN when *k=3* for female class, through the all dimensions is the same, 100%; for male class, through the all dimensions is 80%.

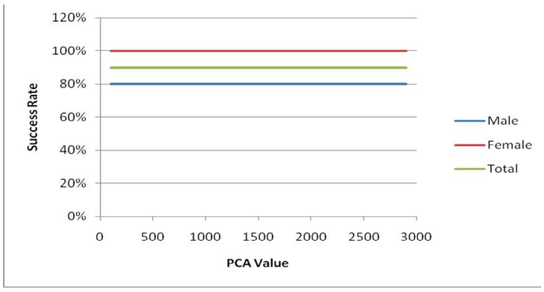

Figure 4. Success ratio of k-NN according to dimension (k=3).

When *k={5,7,9},* the performance of k-NN for each class separately according to reduced dimension is shown in Figure 5. The success ratio of k-NN when *k={5,7,9},* for female class, through the all dimensions is the same, 93%; for male class, through the all dimensions is 87%.

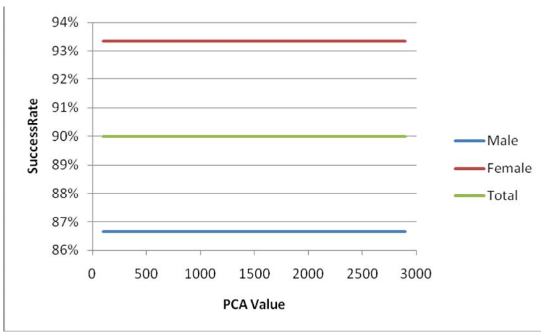

Figure 6. Success ratio of k-NN according to dimension (k={5,7,9}).

The success ratio of MCMGD for each class separately according to reduced dimension is shown in Figure 6. The success ratio of MCMGD for female class, through the all dimensions is 87%; for male class, through the all dimensions is 73%.

These results show that until the dimension reached to 700, SVM did not work efficiently. However after 700, SVM became a stable and efficient classifier for our purpose. It is clearly seen in Figure 2.

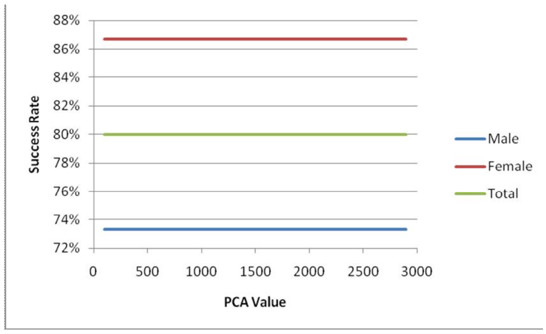

Figure 3. Success ratio of MCMGD according to dimension.

Furthermore, increment of the dimension of the samples does not make big changes in the results of k-NN and Multivariate Classification Method. It is clearly seen in Figure 3-6. Either k-NN or MCMGD approaches gave efficient and stable results for all dimension reduction situations.

Through all algorithms and all dimension reduction experiments, the best performance is obtained by SVM, when number of features is at least 2500.

VI. CONCLUSION

This study has been developed to determine whether a given face image belongs to a man or a woman. Gender recognition could be achieved by several machine learning algorithms. We used SVM, k-NN and Multivariate Classification methods to train and test the system and we gave the comparison of them in the previous chapter. In addition to those, we investigated the effect of the input space's dimension on the performance of the algorithms.

Before learning and testing steps, PCA was used to obtain the main components of images. Through the comparisons we have made; we have realized which methods are effective or ineffective in which situations.

For instance, experimental results illustrated that even if SVM does not show successful results by 700 features, it goes on with a stable and successful way after 700 features and more.

The success of k-NN Method keeps stability regardless of the number of features and also Multivariate Classification shows highly successful and stable results. They are not extremely influenced by the most distinctive features chosen.

The algorithms, used in this study, can be compared by using different data sets. The images of men and women were obtained from various places, such as internet, several image repositories, students and employees in our university

REFERENCES

[1] G.W. Cottrell and J. Metcalfe, "EMPATH: Face, emotion, and gender recognition using holons", *In the Proc. Advances in Neural Information Processing Systems 3 (NIPS)*. Morgan Kaufmann, pp. 564–571.

[2] B.A. Golomb, D.T. Lawrence, and T.J. Sejnowski, "SEXNET: A neural network identifies sex from human faces", *In the Proc.*

Advances in Neural Information Processing Systems 3 (NIPS), Morgan Kaufmann, pp. 572–579.

[3] D. Valentin, H. Abdi, and A.J. O'Toole, "Categorization and identification of face images by neural networks: A review of the linear autoassociative and principle component approaches", *Journal of Biological* Systems, vol.2, pp. 413-429, 1994.

[4] S. Tamura, H. Kawai, and H. Mitsumoto, "Male/female identification from 8 to 6 very low resolution face images by neural network", *Pattern Recognition* 29 (2), 331–335.

[5] L. Wiskottiz, J. Fellous, N. Krüger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching". *In Intelligent Biometric Techniques in Fingerprint and Face Recognition,* eds. L.C. Jain et al., CRC Press, 1999, pp. 355-396.

[6] M. Lyons, J. Budynek, A. Plante, and S. Akamatsu, "Classifying facial attributes using a 2-d Gabor wavelet representation and discriminant analysis", *In: Proc. Internat. Conf. on Automatic Face and Gesture Recognition (FG'00)*, IEEE, Grenoble, France, pp. 202–207.

[7] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", *In: Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR'01*), vol. 1, pp. 511–518.

[8] Z. Sun, G. Bebis, X. Yuan, and S.J. Louis, "Genetic feature subset selection for gender classification: A comparison study", *In: Proc. IEEE Workshop on Applications of Computer Vision (WACV'02*), pp. 165–170.

[9] J. Wu, W.A.P. Smith, and E.R. Hancock, "Facial Gender Classification Using Shape-From-Shading", *Journal of image and vision computing,* vol. 28, June, 2010.

[10] E. Mäkinen and R. Raisamo, "An experimental comparison of gender classification methods", *Pattern Recognition Letters, vol. 29 (2008*), pp. 1544–1556.

[11] H.-C. Lian and B.-L. Lu, "Multi-view gender classification using local binary patterns and support vector machines", *In: Proc. 3rd Internat. Sympos. on Neural Networks (ISNN'06),* Chengdu, China, vol. 2, pp. 202–209.

[12] Y. Saatci, and C. Town, "Cascaded classification of gender and facial expression using active appearance models", *In: Proc. 7th Internat. Conf. on Automatic Face and Gesture Recognition (FGR'06*), pp. 393–400.

[13] B. Moghaddam, and M.-H. Yang, "Learning gender with support faces", *IEEE Trans. Pattern Anal. Machine Intell,* 24 (5), 707-711.

[14] J. Huang, and B. Heisele, "*Component-based Face Recognition with 3D Morphable Models*", Center for Biological and Computational Learning, M.I.T., Cambridge, MA, USA, 2003.

[15] S. Srisuk, K. Ratanarangsank, W. Kurutach, and S. Waraklang, "*Face Recognition using a New Texture Representation of Face Images*", Advanced Machine Intelligence Research Laboratory (AMIRL), 2003.

[16] T.F. Cootes, and C.J. Taylor, "Statistical Models of Appearance for Computer Vision", *Imaging Science and Biomedical Engineering*, University of Manchester, 2000.

[17] M. Turk, and A. Pentland, "Eigenfaces for recognition", *Journal of Cog. Neuroscience*, vol. 3(1), 1991.

[18] E. Alpaydin, *Introduction to Machine Learning*, MIT Press, 2004.

[19] V.N. Vapnik, "An overview of statistical learning Theory", *IEEE Transactions on Neural Networks*, 1999.