Experiments with Eigenedginess: Application to Face Recognition Problem

Hima Patel, Suman K. Mitra
Dhirubhai Ambani Institute of Information and Communication Technology
Gandhinagar, Gujarat, India

Abstract—**Edginess method to extract edges based on one dimensional processing of images is an efficient method for edge extraction.. This paper is an attempt towards answering the following question: How good are features extracted by edginess in terms of their data representation and discrimination properties? We choose face recognition as our testing domain where we use two classifiers in the decision stage: nearest neighbor (NN) and Support Vector Machine (SVM). We observe that there is small difference in the results of these two classifiers with largely different classification abilities, if we extract edginess features apriori. This provides an experimental proof of the fact that edginess feature extraction is a good way to represent discriminating features.**

I. INTRODUCTION

Automatic recognition of faces comes under the general area of object recognition and is a difficult task. It has wide range of applications from security to human computer interface tasks. Researchers have proposed many techniques of extracting features for face recognition purpose. Edginess proposed in [1] to extract the edginess map of an object is a computationally efficient method which uses one dimensional (1D) processing of images. This method has been applied to faces and has shown better performance over other edge and gray level based representations for variation in illumination for human faces [2, 8]. Edginess has also proved to be a good representation in facial expressions classification [7]. Motivated by the promise shown by edginess, it is indeed a curious question whether edginess features have in itself strong representation and discrimination capabilities. In this paper we explore this very fact in the face recognition domain. We extract edginess features of face images, apply Principal Component Analysis (PCA) on these and compare the results of two classifiers namely the SVMs and the nearest neighbor classifier. Section two describes in brief the theory and intuition behind edginess and this work, Section three briefly describes the proposed testing method and Section four gives the experimental results followed by discussion and conclusion in Section 5.

II. ABOUT EDGINESS

Edginess is a strong feature extraction method and has proved to be better than other edge representations [2]. The reason behind this is that edginess is based on one-dimensional processing of images. The traditional 2D operators smooth the image in all directions resulting in the smearing of edge information. To extract the edginess map, the image is smoothed using a 1D Gaussian filter along the horizontal (or vertical) direction to reduce noise. The smoothing filter is a 1D Gaussian filter is given by

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma_1} e^{\frac{-x^2}{2\sigma_1{}^2}} \tag{1}$$

where σ_1 is the standard deviation of the Gaussian function. The response of the 1 D Gaussian filter applied along a particular scan line of an image in one direction. A differential operator (first derivative of 1-D Gaussian function) is then applied in the orthogonal direction, i.e., along the vertical (or horizontal) scan lines to detect the edges. The first order derivative of 1D Gaussian is given by

$$c(y) = \frac{-y}{\sqrt{2\pi}\sigma_2{}^3} e^{\frac{-y^2}{2\sigma_2{}^2}} \tag{2}$$

The resulting image obtained by applying equation 1 produces the horizontal components of edginess (strength of an edge) in the image. Similarly, the vertical components of edginess are derived by applying the above filters on original images in orthogonal directions of those used in obtaining the horizontal components of edginess. Finally the total magnitude of partial edge information obtained in both the horizontal and vertical edge components gives the edginess map of the original image. Figure 1a and 1b show a plot of Gaussian mask and its derivative. Figure 2a-2f shows the various steps in creating an edginess image from a gray scale image. The edginess of a pixel in an image is identical to the magnitude of the gradient of gray level function, which corresponds to the amount of change across the edge. The edginess images of an example face are shown in Figure 2f.

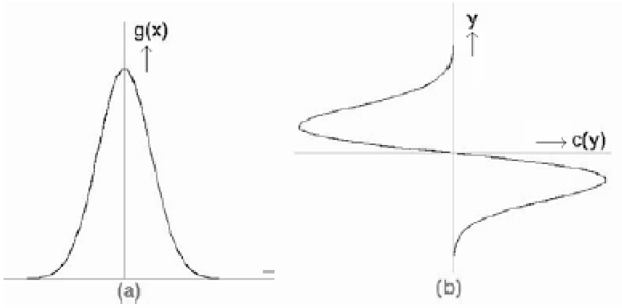

Fig 1. 1(a) Gaussian Function (smoothing filter), 1(b) First derivative of Gaussian (differential operator).

It is visually clear the edginess image carries more information than the edge map of an image. The intuitive reason for this is that the edginess gives a very low output when it operates on completely smooth regions with no useful information. However, unlike the edge detection process, the edginess maintains an output in the regions having even low amount of texture. Again, the 1-d and orthogonal processing of the gaussian and its derivative is less affected by the tradeoff between smoothing out the noise and smoothing the image features. Thus as seen from the face images, the smooth regions of the face that carry no discriminant information, and may cause class overlap in the classification, are removed. However, the regions with even a small amount of discriminant texture are visible in the output. This is the intuitive motivation behind this research, where we need to know whether this information at the output of the edginess filter is really made mainly of the discriminant information of the face.

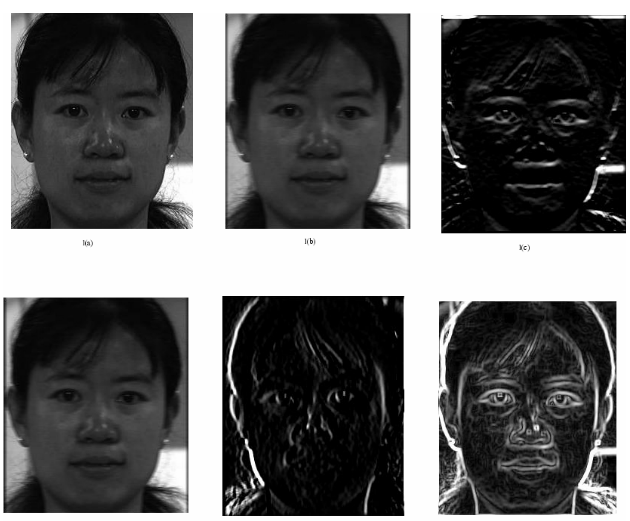

Fig 2. 2(a) Gray Scale Image, 2(b) Image after smoothing in horizontal direction, 2(c) Image after applying the differential operator to 2(b)in vertical direction, 2(d) Image after smoothing in vertical direction, 2(e) Image after applying the differential operator to 2(d) in vertical direction, 2(f) Edginess Image.

III. Proposed Testing Algorithm

To test the above intuitive hypothesis, we follow a simple testing strategy. After finding the edginess map of individual images, PCA is applied on the set of images used for training. Since PCA is applied on the edginess images, the Eigen vector images are referred to as Eigen-Edginess images. The weight vectors obtained from PCA are fed as input to the next stage of classification.

The classification is an important stage that validates our hypothesis. Our argument is as follows: If the edginess features are discriminant enough, then ideally there should be negligible difference between two classifiers of largely varying strength, since even the weaker(e.g. linear) classifier can do as good as a stronger one (non-linear), if the data itself is well-clustered.

Hence, in the classification stage, we apply the classical nearest neighbour classifier and the SVM classifier and compare their results. The NN classifier is a non-parametric, non-trainable, linear classifier and hence has relatively low classification ability. SVM, on the other hand, has its foundations in the elegant statistical theory [3]. SVM employs the Structural Risk Minimization (SRM) principle for optimizing the separating non-linear hyper-plane for better classification. The separating hyper-plane can be trained such that the distance of the closest vectors to the hyper-plane, from the two classes on the opposite side of the hyper-plane is maximized. (Such a hyper-plane is also called a margin). Such an optimization is responsible for high classification accuracy. Another advantage that SVM possesses over the NN classifier is the higher dimensional classification with the use of non-linear kernels in defining the hyper-plane. For example, in a two class case, the hyper-plane is defined as

$$y(x) = \mathrm{sgn}[\sum_{i=1}^{m} w_i k(x.x_i) + b] \qquad (3)$$

where x is the unknown input vector, *wi* are the SVM parameters and b is the bias.

In this case the k defines the non linear kernel. This kernel maps the inputs in the input space to a higher dimensional space. The higher dimensional mapping can represent the non-linear classification in input space as a linear classification in the transformed space. Since the classification is carried out in a higher dimensional space, the classification ability increases. Thus, in cases where the classes tend to overlap as in our case of such a higher order classification will tend to separate the classes further. In our case, the kernel used is the polynomial kernel of degree 2, which is defined as follows,

$$k(x, y) = (1 + x.y)^2 \qquad (4)$$

For a multi class SVM, a one against all classes approach has been used. This method trains k SVM's, where k is the number of classes. The *ith* SVM is trained with all examples from that class as positive and all other examples as negative [5]. Therefore, we then have k decision functions and a test image is classified to a class for which the value of decision function is the largest. A Matlab SVM toolbox has been used for the implementation of SVM classifier [6].

IV. EXPERIMENTAL RESULTS

In order to establish the performance of Edginess-SVM in comparison with Euclidian distance based NN classification, we carried out the experiments on a set of CMU PIE database. All the images considered had a frontal pose and nearly the same expression with wide changes in illumination conditions. We have considered 24 images for one individual and these have been randomly distributed for the training and testing sets. The training and testing sets are so chosen that there is no overlap between them. Different experiments have been performed considering varying number of images for training and the recognition rates have been recorded for both Euclidian distance based NN classification scheme and SVM based classification scheme. The testing set consists of 12 images chosen randomly. Figure 3 shows the comparison of recognition rates between the Eigenedginess-SVM method and Eigenedginess-NN method.

As observed from the graph in Figure 3, the performance of both NN and SVM is nearly same except for the cases when the number of training images is less. However, in [4] the authors have shown that classification by SVM's is more efficient than that by nearest neighbor scheme for face recognition problem with PCA as the feature extraction technique. The reason for this is that nearest neighbor based scheme is very sensitive to noisy inputs and can easily get confused with the neighboring classes in the eigen space. The latter is due to the fact that it does not perform classification based on discriminatory function like in SVM's but on the data points itself. This leads us to conclude that possibly edginess is a strong feature extraction method and the classification is not affected much by the classifier used at the back end.

V. CONCLUSION

This work presents a hypothesis that the features extracted by edginess are indeed very informative about the discriminative elements of the face image. The experiment verifies the above hypothesis not only for normal frontal faces but also in cases of wide illumination variation. This leads us to comment that edginess features do not possess only good discriminative property but also illumination invariance. That is, the features edginess represents, maintain their discriminative property even under wide illumination changes. Hence, using edginess in the prior stage of feature extraction, we can do with simple weaker classifiers, obviating the use of more complicated and better classifiers.

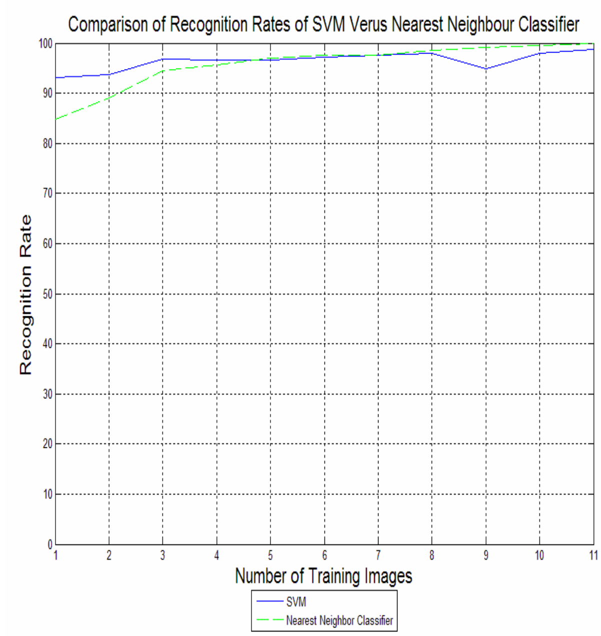

Fig 3. The graph shows a comparison of recognition rate between SVM and NN classifier for different number of training images.

REFERENCES

[1] P. Kiran Kumar, Sukhendu Das and B. Yegnanarayana, "One Dimensional processing of images", in *Int. Conf. on Multimedia processing Systems, Chennai, India*, Aug. 13-15, 2000., pp 181-185.
[2] S Ramesh,S Palanivel, Sukhendu Das and B Yegnanarayana, "Eigenedginess vs. eigenhill, eigenface and eigenedge", *In Eleventh European Signal Processing Conference*, 2002, pp 559-562.
[3] V.Vapnik, "An Overview of Statistical Learning Theory", *IEEE Transactions on Neural Networks*, 1999, pp 988-999.
[4] Guodong Guo, Stan Z Li, Kap Luk Chan, "Support Vector Machine for Face Recog-nition", *Image And Vision Computing, Elsevier*, 2001, pp 631-638.
[5] Chih-Wei Hsu and Chih-Jen Lin, "A Comparison of Methods for Multi Class Sup-port Vector Machines", *IEEE Transactions on Neural Networks*, 2002, pp 415-425.
[6] S. Canu, Y. Grandvalet, V. Guigue, A. Rakotomamonjy, "SVM and Kernel Methods Matlab Toolbox", *Perception Systems Information*, 2005.
[7] A Bhavsar and H Patel. Facial expression recognition using neural classifier and fuzzy mapping. *IEEE Indicon 2005 Conference*, 2001, pp 383-387.
[8] K. Srinivasa Rao, A. N. Rajagopalan, “A Probabilistic Fusion Methodology for Face Recognition”, *EURASIP Journal on Applied Signal Processing,* 2005:17, 2772–2787.