International Journal of Electronics Engineering Research.
ISSN 0975-6450 Volume 8, Number 1 (2016) pp. 1-8

http://www.ripublication.com

Face Recognition using SURF Features and SVM Classifier

Bhaskar Anand
Student, M. Tech, Department of Electronics Engineering,
SVNIT, Surat, Gujrat-395007, India
E-mail: bhaskaranand.nitjsr@gmail.com

Mr. Prashant K Shah
(Associate Prof.) Department of Electronics Engineering,
SVNIT, Surat, Gujarat-395007, India
E-mail: pks@eced.svnit.ac.in

Abstract

Face recognition is a fascinating research topic in recent years. Numerous methods and algorithms have been suggested by researchers. The accuracy of face recognition technique is affected by factors like variation in illumination, facial expression, scaling and perspective movement. It is important to note that Speeded-up Robust Features (SURF) extracted from a facial image are invariant to shifting, scaling and rotation. In addition to that they are also partially invariant to illumination and affine transformation. This paper suggests a facial recognition technique using SURF features and Support Vector Machine (SVM) classifier. Above techniques has been tested on Yalefaces and UMIST face databases. The results indicate that the proposed method can lead to high recognition efficiency..

Keywords: Face feature, face recognition, SURF feature, SVM

INTRODUCTION

Face recognition has been researched from last many decades. Face recognition aims at recognizing a given image from an image database. For this purpose we need to first determine the facial area (face detection). Then some well defined features are detected (Feature detection) from the image. These features need to be represented as vectors having numerical values (Feature extraction) so that it can later be used to match (Classification) with feature vector of other image.

There are many impediments in achieving high recognition rate. Some of them are change in expression, illumination, rotation and scaling. To deal with these problems some local features have been introduced by researchers. Local features such as corners and blobs are mostly used for object recognition. Scale Invariant Feature Transform (SIFT) introduced by D. Lowe [1] [2] has been widely used in the field of Face recognition. However since its computing time is high it limits the speed in live image applications. Speeded-Up Robust Feature (SURF) [3] [4] introduced by H. Bay et al in 2008 provides all the benefits of SIFT such as scale invariance, illumination invariance etc. but at lower computing time. The size of SURF feature vector (64) is also smaller as compared to SIFT (128) which helps in faster classification.

Support Vector Machine (SVM) is a widely used method for data classification as well as regression. We have used SURF features as features of image as SVM as classifier. But instead of doing classification of face images as a whole we have done feature wise classification to decide label of the image. Rest of the paper proceeds as follows; second part is an introduction to SURF and SVM. Third part describes the suggested method. Fourth part shows the results and fifth part is conclusion.

INTRODUCTION TO SURF AND SVM

SURF

SURF is a local feature detector and descriptor that can be used for tasks such as object recognition or registration or classification or 3D reconstruction. It is a scale and in-plane rotation invariant feature. The detector locates the interest points in the image, and the descriptor describes the features of the interest points and constructs the feature vectors of the interest points. SURF features are invariant of shifting, rotation and scaling, and partially invariant to illumination and affine transformation.

$$H(x,\sigma)=\begin{pmatrix} L_{xx}(X,\sigma) & L_{xy}(X,\sigma) \\ L_{xy}(X,\sigma) & L_{yy}(X,\sigma) \end{pmatrix} \qquad (1)$$

where $L_{xx}(X,\sigma)$ is the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial x^2} g(\sigma)$ with the image I in point x, and similarly for $L_{xy}(X,\sigma)$ and $L_{yy}(X,\sigma)$.

The heart of the SURF detection is non-maximal suppression of the determinants of the hessian matrices. The convolution is very costly to calculate and it is approximated and speeded-up with the use of integral images and approximated kernels [3].

The second order Gaussian kernels $\frac{\partial^2}{\partial x^2} g(\sigma)$ used for the hessian matrix must be discretized and cropped before we can apply them, a 9x9 kernel is illustrated in Figure 1. The SURF algorithm approximates these kernels with rectangular boxes, box filters.

For scale invariance, the SURF constructs a pyramid scale space, like the SIFT. Different from the SIFT to repeatedly smooth the image with a Gaussian and then sub-sample the image, the SURF directly changes the scale of box filters to implement the scale space using box filter and integral image. Next step is interest point description. SURF uses the sum of the Haar wavelet responses to describe the feature of an interest point [2]. Fig. 2 shows the Haar wavelet filters used to compute the responses at x and y directions. For the extraction of the descriptor, the first step consists of constructing a square region centered at the interest point and oriented along the orientation decided by the orientation selection method introduced in [2].

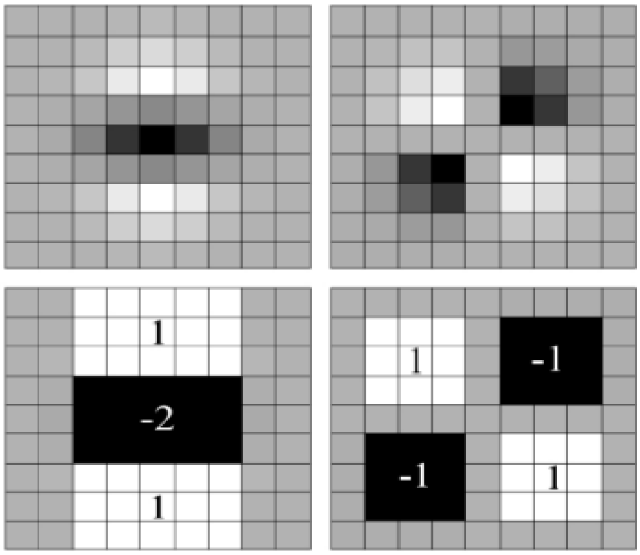

Figure 1: Discretized and cropped Gaussian second order partial derivatives in y-direction and xy-direction, and their approximations using box filter [3]

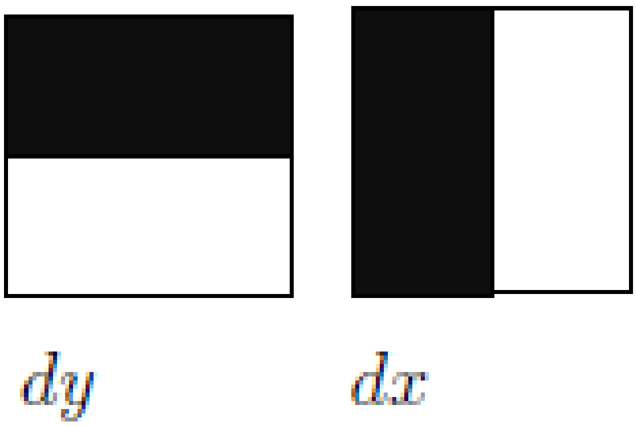

Figure 2: The wavelets response. Black and white areas corresponds to a weight-1 and 1 for the Haar kernels [3]

The region is split up equally into smaller 4x4 square subregions (as shown in Fig.3). This preserves important spatial information. For each sub-region Haar wavelet responses are computed at 5x5 equally spaced sample points. Haar wavelet response in horizontal direction is referred as dx and dy the Haar wavelet response in vertical direction. To increase the robustness towards geometric deformations and localization errors, the responses dx and dy are first weighted with a Gaussian centered at the interest point.

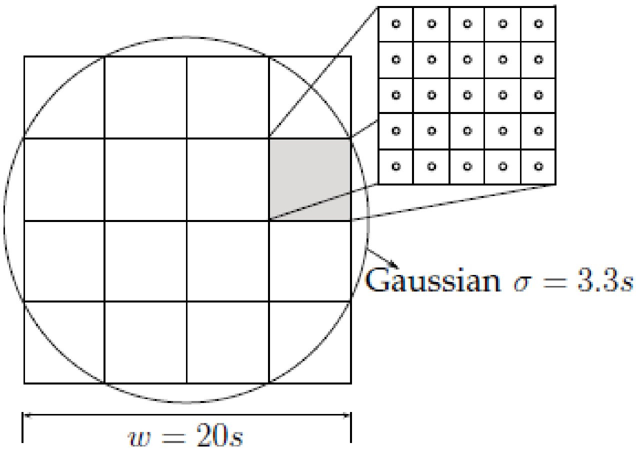

Figure 3: The 20s areas are divided into 4X4 subareas that are sampled 5X5 times to get the wavelet response [3]

SVM

Now we should have a brief introduction of SVM. Support Vector Machine (SVM) is primarily a classifier that performs classification tasks by constructing hyperplanes in a multidimensional space separating cases of different class labels. According to SVM the decision boundary should be as far away from the data of both classes as possible. Let us consider that we have data points belonging to two classes, A_+ and A_-. Each point in the dataset comes with a class label y, +1 or 1, indicating one of two classes A_+ and A_-.

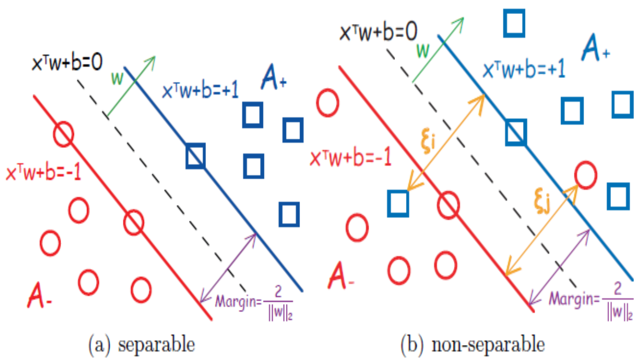

Figure 4: Linear SVM for separable and non separable data [5]

Let us start with a strictly linearly separable case, i.e. there exists a hyperplane which can separate the data A_+ and A_-. In this case we can separate the two classes by a pair of parallel bounding planes (eq.2) [5]:

$$\begin{aligned} w^T x + b &= +1 \\ w^T x + b &= -1 \end{aligned} \qquad (2)$$

Where w is the normal vector to these planes and b determines their location relative to the origin. The first plane of (eq. 2) bounds the class A_+ and the second plane bounds the class A_-. SVM achieves better prediction ability via maximizing the margin between two bounding planes. Hence, SVM searches for a separating hyperplane by maximizing $\frac{2}{\|w\|_2}$ The linear separating hyperplane is the plane

$$w^T x + b = 0 \qquad (3)$$

By maximizing the margin between the bounding planes we get an optimal solution (w^*, b^*). The data points on the bounding planes, $w^{*T}\ x + b^* = \pm 1$, are called support vectors. Once we have the training result, all we need to keep in our databases are the support vectors. If the classes are linearly inseparable then the two planes bound the two classes with a soft margin [5] [6] determined by a nonnegative slack vector variable ξ, that is:

$$\begin{aligned} w^T x_i + b + \xi_i &\geq +1, \text{for } x_i^T \in A_+ \\ w^T x_i + b - \xi_i &\leq -1, \text{for } x_i^T \in A_- \end{aligned} \qquad (4)$$

Many datasets cannot be well separated, even after using a soft margin, by a linear separating hyperplane, but could be linearly separated if mapped into a higher or much higher dimensional space by using a nonlinear map. Rather than mapping individual features to higher dimension. Kernel method is used for this purpose. The kernel computes the dot product which would otherwise be much more expensive to compute explicitly. Some widely used Kernel functions are Linear, Polynomial, Radial Basis Function (RBF) and Multi-Layer Perceptron (MLP). We will be using RBF which is given by:

$$k(x, z) = \exp\left(-\frac{\|x - z\|^2}{2\gamma^2}\right) \qquad (5)$$

The parameter gamma has to be chosen carefully for best results of classification.

PROPOSED METHOD

Figure 5 shows the general block diagram of the proposed method. In our approach SURF features of face images are extracted and matched using SVM classifier. In this approach we take few training images to train SVM classifier. A database is created having facial SURF features of training images of all the individuals. For a new image (person), first of all the facial region will be detected (if it is not a cropped face image).For detecting face region we use Viola-Jones algorithm for face detection [7]. Then the SURF features are detected for the face region. Output of this step is SURF

Points object, containing information about SURF features detected in the 2-D grayscale input image. After this step feature extraction will be done. This step returns extracted feature vectors, also known as descriptors, and their corresponding locations. This feature vector will have a size of nx64, where n is the number of extracted feature points. Now this feature vector will be classified using SVM classifier.

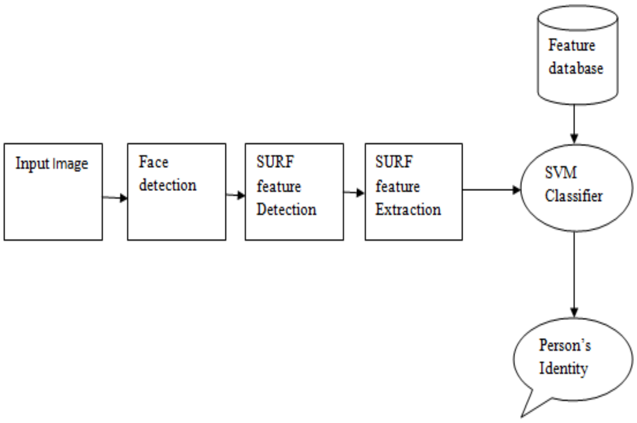

Figure 5: Proposed method for face recognition

Now while doing classification, we have trained our classifier not the individual face images as a whole but in terms of the features. For example rather than labeling a particular image as r, where r is an integer, we have labeled its features The kernel computes the dot product which would otherwise be much more expensive to as r. Thus if we get n features from a sample then n rows (feature vector) will be labeled as r. Similarly while testing also rather than predicting label of testing image as a whole labels of obtained features are predicted. If the label of majority of features is matching with that of expected label then its a successful recognition. We have used RBF kernel for classification. We optimized parameter gamma and C of RBF kernel by Cross-validation and Grid-Search method as proposed by Chih-Wei Hsu et al [8] [9].

RESULTS

We need multi-class classification as we will always have many classes (the face images of different people). Simulation has been done on Matlab. We used Libsvm, a library for Support Vector Machines for performing multi-class SVM. For testing our method we have used standard face databases Sheffield (previously UMIST) and Yalefaces. The Sheffield Face Database consists of 564 images of 20 individuals. The images of each individual cover a range of poses from frontal towards side view and illumination changes as well. This database is suitable to test perspective change and illumination invariance. The Yalefaces database contains 165 images of 15 individuals with different facial expression. For Sheffield and Yalefaces databases we used 10 and 5 images per person respectively for training our classifier. Sample images of the two databases are shown in figure 6 and 7. The result is summarized below in tables I and II.

Figure 6: Yalefaces face image database sample images

Figure 7: UMIST face image database sample images

Table I: accuracy with yalefaces and umist database

Database	Yalefaces	UMIST
Accuracy	97.78	97.87

Table II: accuracy with different scaleswith umist database

Scale	1	2	0.75	0.50
Accuracy	97.87	93.60	93.86	87.46

CONCLUSION AND FUTURE WORK

SURF features provide obvious advantage of invariance of shifting, rotation and scaling, and partial invariance to illumi nation and affine transformation. Almost same advantages are also provided by SIFT features. But SURF is much faster than SIFT with respect to feature detection and feature extraction. The Support Vector Machine (SVM) classifier is a powerful classifier that works well on a wide range of classification problems, even problems in high dimensions and that are not linearly separable. SVM is suited in the cases when we need to deal with very high dimensional data. In our case each feature is of 64-dimension. Then each class has many features. Hence overall we get a high dimensional feature dataset. In such cases SVM can be a good alternative. Results show a good accuracy with the images having variation in illumination, perspective movement, facial expression and scaling. In future we can also think of including image preprocessing so for better illumination and scaling invariance.

REFERENCES:

[1] D. Lowe, ”Distinctive image features from scale-invariant keypoints, cascade filtering approach”, International Journal of Computer Vision, vol 60, pp 91-110, January 2004.

[2] Lowe, D.G. 1999. Object recognition from local scale-invariant features. In International Conference on Computer Vision, Corfu, Greece, pp. 1150-1157.

[3] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, ”Speeded-up robust features (SURF)”, Comput. Vis. Image Underst., 110(3), 346-359 (2008).

[4] A technical report on Feature detection and implementing the Speeded-Up Robust Features(SURF) algorithm, available at http://cs.au.dk/ jtp/SURF/report.pdf

[5] Yuh-Jye Lee, Yi-Ren Yeh, and Hsing-Kuo Pao, An Introduction to Support Vector Machines, National Taiwan University of Science and Technology, Taipei, Taiwan

[6] Chih-Chung Chang, Chih-Wei Hsu, and Chih-Jen Lin. The analysis of decomposition methods for support vector machines. IEEE Transactions on Neural Networks, 11(4):10031008, 2000.

[7] M. Jones and P. Viola, Face Recognition Using Boosted Local Features, IEEE International Conference on Computer Vision, 2003.

[8] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, A Practical Guide to Support Vector Classification, Department of Computer Science, National Taiwan University, 2010

[9] C.-C. Chang and C.-J. Lin., ”LIBSVM: a library for support vector machines, 2001, available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.