

FACE DETECTION AND IDENTIFICATION USING SVM

Sushma Ronanki¹, Sonia Gundu², Rupavathi Baratam³,
P Mounika⁴, J Rajesh Kumar⁵

¹Assistant Professor, M.Tech, ^{2,3,4,5}Student, B.TECH, Electronics and Communication,
SSCE, Srikakulam, Andhra Pradesh (India)

ABSTRACT

In the real-time scenario, the concept of face recognition has becoming an essential focus in research/study communities. The main problem is to extract facial features from a large complete data set of images. In our proposed algorithm, a trained classifier is used to classify the complete feature vectors obtained from the image set taken and the features of query image is compared with this trained set so as to obtain a complete face recognition system with higher accuracy.

Keywords: classifier, face recognition, facial features, MATLAB.

I. INTRODUCTION

The demand for security of assets as well as for some private information without losing some people identity makes to concentrate on a system set to be biometrics. Different types of biometric systems are fingerprint recognition, speaker recognition, iris recognition, face recognition and signature verification. Over the last ten years or so, face recognition has become a popular area of research in computer vision and one of the most successful applications of image analysis and understanding. Because of the nature of the problem, not only computer science researchers are interested in it, but neuroscientists and psychologists also. It is the general opinion that advances in computer vision research will provide useful insights to neuroscientists and psychologists into how human brain works, and vice versa. The goal is to implement the system (model) for a particular face and distinguish it from a large number of stored faces with some real-time variations as well. It gives us efficient way to find the lower dimensional space. Further this algorithm can be extended to recognize the gender of a person or to interpret the facial expression of a person. Recognition could be carried out under widely varying conditions like frontal view, a 45° view, scaled frontal view, side view is tried, while the training data set covers limited views. The algorithm models the real-time varying lighting conditions as well. But this is out of scope of the current implementation. The aim of this research paper is to study and develop an efficient MATLAB program for face recognition using support vector machine and to perform test for program optimization and accuracy.

II. FACE RECOGNITION:

Face recognition is classified into two types. They are

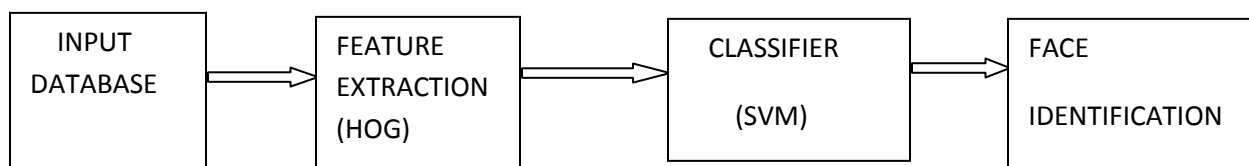
- 1.face verification
- 2.face identification

Difference between Face Verification and Face Recognition identification:

Face Authentication (also called verification) involves a one to one check that compares an input image (also called a query image, probe image or simply probe) with only the image (or class) that the user claims to be. In simple words, if you stand in front of a face authentication system and claim to be a certain user, the system will ONLY check if you are that user or not.

Face Recognition (or Identification) is another thing, though of course related. It involves a one to many comparison of the input image (or probe or query image) with a template library. In simple words, in a face recognition system the input image will be compared with ALL the classes and then a decision will be made so as to identify to WHO the input image belongs to. Or if it does not belong to the database at all.

Face recognition can be achieved by the following steps:



III. INPUT DATABASE

It is easy to identify a person from a group of 5 or 6 people whereas identifying a person from a group of more than thousand people will be difficult. Here in the input data base several images were taken. with the help of facial features of those images identification of a person is done.

IV. FEATURE EXTRACTION:(HOG)

For extracting the features of an image here HOG is used. HOG stands for histogram of oriented gradient. Feature extraction starts from initial set of measured data and builds derived features. when the input data to an algorithm is too large to be processed and it is suspected to be redundant then it can be transformed into a reduced set of features.

Histogram of oriented gradients (HOG) is a feature descriptor used to detect objects in computer vision and image processing. The HOG descriptor technique counts occurrences of gradient orientation in localized portions of an image - detection window, or region of interest (ROI).

Implementation of the HOG descriptor algorithm is as follows:

1. Divide the image into small connected regions called cells, and for each cell compute a histogram of gradient directions or edge orientations for the pixels within the cell.
2. Discretize each cell into angular bins according to the gradient orientation.
3. Each cell's pixel contributes weighted gradient to its corresponding angular bin.

4.Groups of adjacent cells are considered as spatial regions called blocks. The grouping of cells into a block is the basis for grouping and normalization of histograms.

5.Normalized group of histograms represents the block histogram. The set of these block histograms represents the descriptor.

The essential thought behind the histogram of oriented gradients descriptor is that local object appearance and shape within an image can be described by the distribution of intensity gradients or edge directions. The image is divided into small connected regions called cells, and for the pixels within each cell, a histogram of gradient directions is compiled. The descriptor is the concatenation of these histograms. For improved accuracy, the local histograms can be contrast-normalized by calculating a measure of the intensity across a larger region of the image, called a block, and then using this value to normalize all cells within the block. This normalization results in better invariance to changes in illumination and shadowing.

The HOG descriptor has a few key advantages over other descriptors. Since it operates on local cells, it is invariant to geometric and photometric transformations, except for object orientation. Such changes would only appear in larger spatial regions. Moreover, as coarse spatial sampling, fine orientation sampling, and strong local photometric normalization permits the individual body movement of pedestrians to be ignored so long as they maintain a roughly upright position. The HOG descriptor is thus particularly suited for human detection in images.

V. CLASSIFIER :(SVM)

Face recognition systems are definitely still an active area of research.

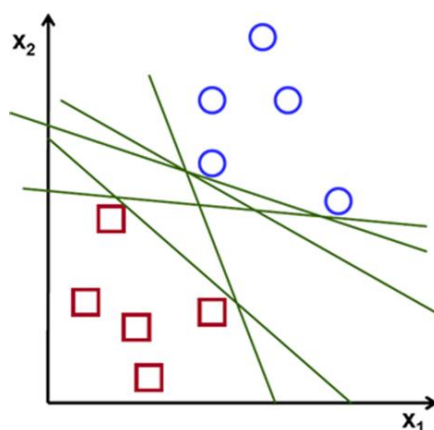
The support vector machine algorithm invented by Vapnik (1998) has been effectively applied to many pattern recognition problems.

In this we are using support vector machine (SVM) as a classifier.

What is a SVM?

A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes new examples.

For a linearly separable set of 2D-points which belong to one of two classes, find a separating straight line.

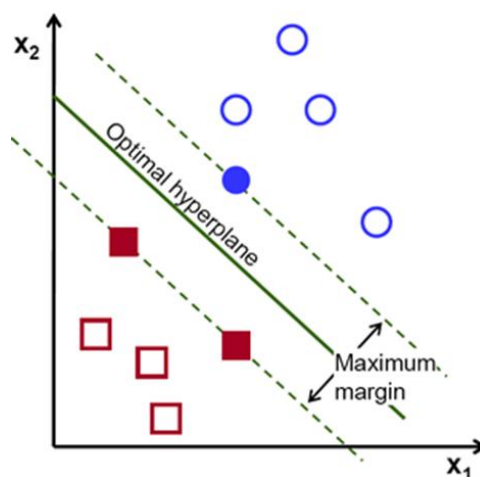


Note: In this example we deal with lines and points in the Cartesian plane instead of hyperplanes and vectors in a high dimensional space. This is a simplification of the problem. It is important to understand that this is done only because our intuition is better built from examples that are easy to imagine. However, the same concepts apply to tasks where the examples to classify lie in a space whose dimension is higher than two.

In the above picture you can see that there exists multiple lines that offer a solution to the problem. Is any of them better than the others? We can intuitively define a criterion to estimate the worth of the lines:

A line is bad if it passes too close to the points because it will be noise sensitive and it will not generalize correctly. Therefore, our goal should be to find the line passing as far as possible from all points.

Then, the operation of the SVM algorithm is based on finding the hyperplane that gives the largest minimum distance to the training examples. Twice, this distance receives the important name of margin within SVM's theory. Therefore, the optimal separating hyperplane maximizes the margin of the training data.



The training points closest to the computed separating hyperplane are called support vectors. The original SVM is a linear binary classifier, which is useful for two-class classification problems. On the other hand, it does not provide good separation for non-sparse complex data (e.g., image data). We will now shortly describe the workings of the SVM.

A Support Vector Machine (SVM) performs classification by constructing an N-dimensional hyperplane that optimally separates the data into two categories. SVM models are closely related to neural networks. In fact, a SVM model using a sigmoid kernel function is equivalent to a two-layer, perceptron neural network.

Support Vector Machine (SVM) models are a close cousin to classical multilayer perceptron neural networks. Using a kernel function, SVM's are an alternative training method for polynomial, radial basis function and multi-layer perceptron classifiers in which the weights of the network are found by solving a quadratic programming problem with linear constraints, rather than by solving a non-convex, unconstrained minimization problem as in standard neural network training.

In the parlance of SVM literature, a predictor variable is called an attribute, and a transformed attribute that is used to define the hyperplane is called a feature. The task of choosing the most suitable representation is known as feature

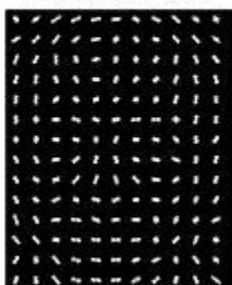
selection. A set of features that describes one case (i.e., a row of predictor values) is called a vector. So the goal of SVM modeling is to find the optimal hyperplane that separates clusters of vector in such a way that cases with one category of the target variable are on one side of the plane and cases with the other category are on the other side of the plane. The vectors near the hyperplane are the support vectors.

VI. EXPERIMENTAL RESULTS:

Input Face



HoG Feature

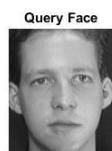


RESULT1:IN THE ABOVE FIGURE THE HOG FEATURES OF A QUERY IMAGE ARE EXTRACTED.



RESULT-2:EXTRACTING QUERY IMAGE FROM
A LARGE DATASET

RESULT-3



RESULT-4

RESULT-5



RESULT-6

RESULT-7

VII. CONCLUSION

The face detection is an active research area, in the last years there were great advances in developing algorithms that solve face recognition problems. The support vector machine is a powerful tool to solve problems and can be applied at each stage of a face recognition system. Face recognition technologies have been associated generally with very costly top secure applications. Certain applications of face recognition technology are now cost effective, reliable and highly accurate. This approach is more appropriate while comparing with other approaches in order to provide improved generalization. This method is fast and convenient in identifying a person.

REFERENCES

- [1.] M. Owayjan, A. Dergham, G. Haber, N. Fakih, A. Hamoush, and E. Abdo "Face Recognition Security System," in New Trends in Networking, Computing, E-Learning, Systems Sciences, and Engineering, Lecture Notes in Electrical Engineering 312, K. Elleithy and T. Sobh, Eds., 1st ed., Switzerland: Springer International Publishing, 2015, pp. 343-348.
- [2.] C. Wallraven, B. Caputo, A. Graf, "Recognition with local features", Proc. of ICVV IEEE 2003.
- [3.] Z. Zeng, Y. Fu, G. I. Roisman, Z. Wen, Y. Hu, T.S. Huang, "Spontaneous emotional facial expression detection", J. Multimedia, Vol. 1, Aug 2006.