International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 3, March - 2013

Illumination And Pose Variation Across Face Recognition Using Local Mapping Analysis

Vidin Sujith M, Vinoth R, Sathishkumar M

Abstract

Face recognition is one of the biometric based applications for identifying or verifying the user by his/her faces. Mainly the face recognition can be done by using algorithms for computing as well as processing the face for specific purpose. In this face recognition system is a computer application for automatically identifying a person from a video frame from a video as live. In our project, we using concept of neural network to develop a novel algorithm called Local Mapping Analysis. This algorithm leads to eliminate the several problem occurs in recognize the face like illumination variation, pose variation etc. This algorithm is also extracting some unique feature like distance between eye, shape of the cheekbones, and the length of the jaw line etc.

Keyword: Local mapping analysis, Illumination and Pose variation

1. Introduction

Face recognition is a computer-vision based application for security purpose. This face-recognition use the face as input to check the face in the database and it produce two results like "match" and "mismatch" [1]. Basically using face detection as the primary step for any kind of face recognition. Face detection is used in finding whether the given image contains a face or not. If present, return the face as output otherwise display as "no face detected".

Recognition is the next step of face detection the detected faces are used as input of this step [2]. But recognition process is not easy because of many problems like age, illumination variation, pose, occlusion and expression. A general face recognition system includes two components, namely face image processing and face recognition algorithm. They correspond to the digital. In our project, we use the improved skin color model algorithm for detecting faces in the image. Basically, the skin color model uses only the RGB color for detection. But improved skin color model using the CYMK color for detecting the face. So it is more accurate than older methods. This project also introduces the nove1 algorithm named as Local mapping analysis. This algorithm analysis the local features of the faces like length of eyes, width of nose etc.. It is one of the neural network based algorithm and trains the set of image as dataset. In this algorithm we convert the colored image into black and white (gray scale) image for processing. This grayscale image is only used for face recognition in any kind of system. This image read by using the imread function and it transform as matrix value. This matrix value is taken as competition parameter. Using the formula we can calculate the features of the face in the matrix.

This algorithm is extracting the features from the face and also stored in the database. This algorithm can do some filtering process for eliminate the unwanted data form or surrounded from the faces. The filtering process is smoothing the template results [4].

Using the ID we can save the feature points for particular person. Each ID has an indent template to the nodal value i.e. feature extraction point value to be saved. In comparison function, it also extracts the feature of the face and compares with database. This function only used to determine whether the person is authorized or not.

Figure 1: Various nodal points

2. Proposed work:

In this project to implement and design the automated face recognition system can be sub divided into four modules. The first module is detecting the face in the given image using improved skin color model as the input of face recognition. The second module is using the local mapping analysis algorithm for extracting the various features of the face and this module is responsible for storing the nodal points as template in the database.[2]

The third module is used to comparing the input face with the database as per the algorithm. Fourth module is only display the message as authenticated person or not by using several conditions in local mapping analysis.

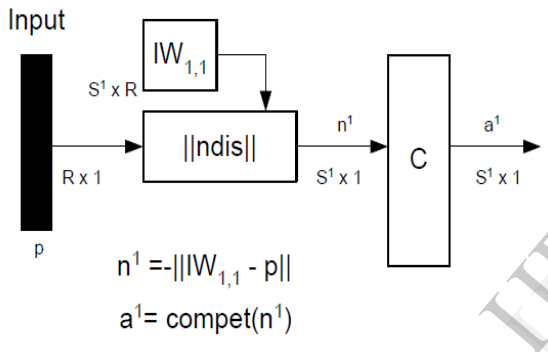

Figure 3: Architecture

Input vector p is the pixel of image. The architecture accepts the input p and input weight W produce a S1 vector. The elements are used to find the distance between the input and weight vector [3]. The competitive transfer function accepts the new input vector for processing and it said to be winning neuron.

3. Algorithm: Local Mapping Analysis

A local mapping analysis is a type of artificial neural network that is used to trained image using unsupervised learning and it produce a low-dimensional, discredited representation of the input of the training samples are called a map [5].

It is different from other artificial neural networks because it uses a neighbourhood function to preserve the topological properties of the input space as shown in fig.

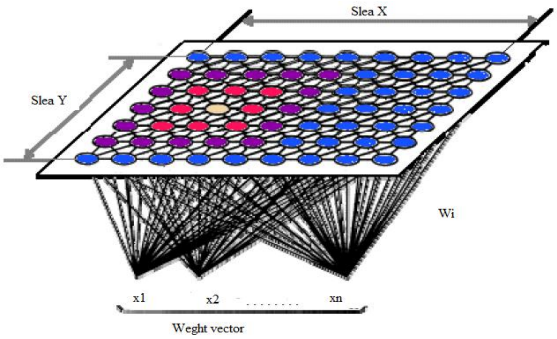

Figure 2: Local Mapping Analysis

Use the feature mapping for grouping the data with/without knowing the class of the input data [1]. This can be used to detect in the test image. After finding the face, cropped as well as save in a dir as shown in figure 1

Figure 4: Crop the face from image

Input: Training data: Face image are taken as x–vectors of length n data

$$\left.\begin{array}{l}(x_{1,1}, x_{1,2}, \ldots, x_{1,i}, \ldots, x_{1,n})\\(x_{2,1}, x_{2,2}, \ldots, x_{2,i}, \ldots, x_{2,n})\\\ldots\\(x_{j,1}, x_{j,2}, \ldots, x_{j,i}, \ldots, x_{j,n})\\\ldots\\(x_{p,1}, x_{p,2}, \ldots, x_{p,i}, \ldots, x_{p,n})\end{array}\right\}\ \text{P distinct training vector}$$

Outputs:

Display the message like the person is authenticated or unauthenticated depends on results of the face recognition.

Steps:

In this implemented by eight following steps,

Step 1: Choosing the random values for the weight vector wj(0), and the weight values begin from 0,1,2,…n. where n can be depends on number of nodal points are taken from face image as shown in fig.

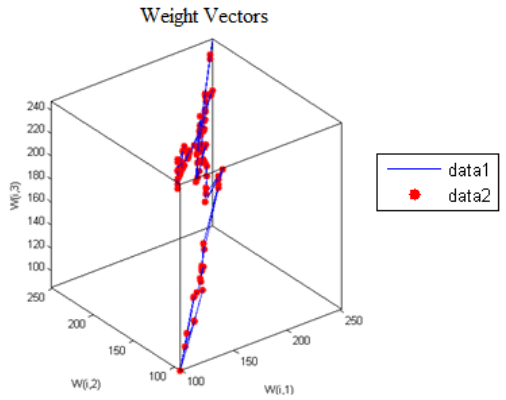

Figure 5: Local Mapping Analysis Weight Vector

Step 2: Find the Euclidean distance to compute the relationship between the adjacent pixel and store in variable wj(n) as following expression,

$$\sum_{k=1}^{n}\left(i_{l.k} - w_{j,k}(t)\right)^2$$

Step 3: Finding the best neuron i(x) by using the minimum Euclidean distance criterion

$$i(x) = \arg\min\|x(n) - wj\|,\ j = 1,2,...,l$$

Step 4: A feature map network identifies a winning neurons. Update only the winning neuron instead of all neurons within a certain neighbourhood of winning neuron. Specifically, we adjust all such neurons as follows:

$$_i w(q) =_i w(q-1) + \alpha(p(q) -_i w(q-1))$$

Here the neighbourhood $Ni(d)$ contains the indices for all the neurons.

$$N_i(d) = \{j, d_{ij} \le d\}$$

Formulas for updating the winning neuron are

$$Wj\ (n+1) = wj\ (n) + f\text{Ø}(n)hj,\ i(x)\ (n)(x(n) - wj\ (n))$$

Where,

fØ(n) is the learning rate parameter, and hj, i(x)(n) is the neighborhood function centered around the winning neuron i(x). Both fØ(n) and hj, i(x) (n) are varied dynamically during learning for best results.

Step 5: Repeat the step 2, step 4 and step 3 until no changes in the weight vector are observed.

Step 6: Using the weight vector to mapping a graph G with m nodes. We put a node I and J and edge xi and xj respectively as shown in figure 4

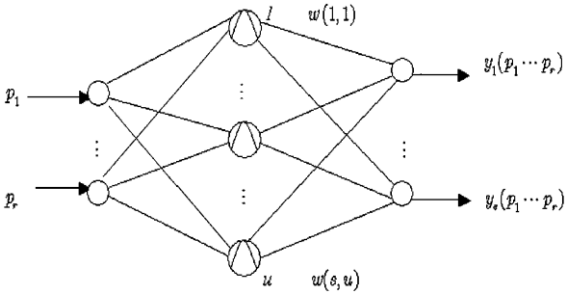

Figure 6: Graph of the weight vector

Step 7: Choosing the weight to an edge. W is a spare symmetric matrix mm with joining edge of I and J and also 0 denotes no edge found as following equation,

$$d1(p,q) = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(p_i - q_i)^2}$$

Where,

M- Dimension of feature vector

Pi- Database feature vector

Qi- test feature vector

Step 8: computing the Eigen values and Eigen vector of a matrix obtained from the graph G as formulated as follow,

$$XLX^T\mathbf{a} = \lambda XDX^T\mathbf{a}$$

Where,

D is the diagonal of the matrix whose column entries are sum of W, is Dii = Σ jWji. And L = DW.

In this algorithm allow to save an image with spatial relationship of the pixel in ORL dataset are show in figure 7. These algorithms are establishing a relationship among the various images to save in database.

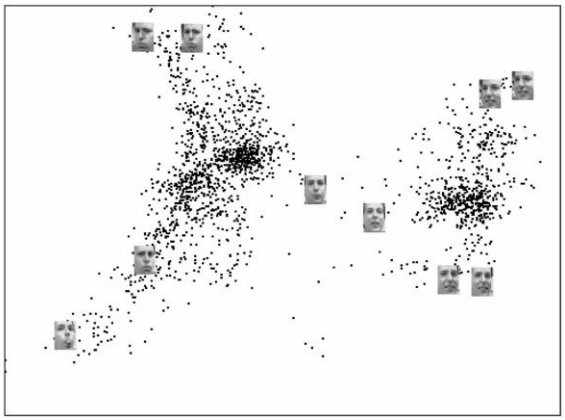

Figure 7: Distribution of the 10 testing samples in the reduced representation subspace

4. Experiment

In this project can be tested by two dataset called ORL and YALE dataset.

ORL database:

The ORL dataset contain 40 subjects and each subject has 10 different poses of the person. It shows the face variation as facial expression, facial detail and poses. All images are gray scale image and original sizes of images are 112 x 92. And it can be resized as 64 x 64 for computational purpose [3]. We experiment the FAR and FRR rate versus threshold value of my algorithm as shown in below graph

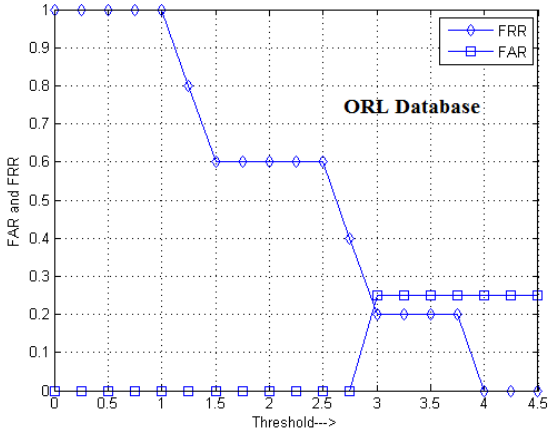

Figure 8: ORL database

Yale database is an international database which contains more than 400 images of an individual. It is gray scale image and it size of 128 x128 dimensional. The FRR and FAR values are varies from changing the threshold value of the algorithm.

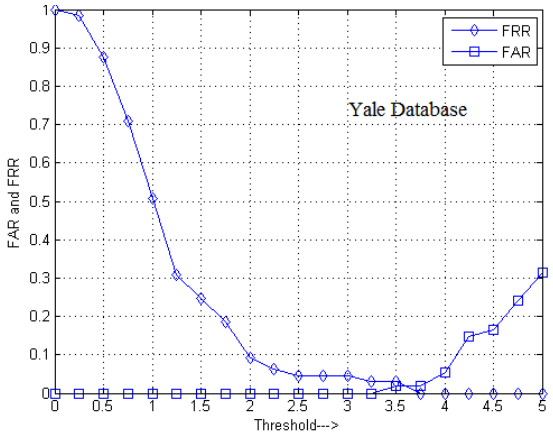

Figure 9: Yale Database

6. Conclusion

This project are mainly focused on solving the existing problem like illumination variation and pose variation using novel algorithm called Local Mapping analysis. This project mostly applicable in road side application like video surveillance etc.., It is an image based neural network in face recognition. This project applicable in windows as well as Linux based application for security purpose. We hope this project to solve the security problem as well as replacing the conventional method.

7. Reference

[1] M.H. Hayes and A.U. Batur, "Linear Subspace for Illumination Robust Face recognition," Dec. 2001.
[2] P.N. Bellhumeur, J.P. Hespaasnha, and D.J. Kriegman, "Fisher-faces vs Eigen-faces: Recognition Using The Class Specific and Linear Projection,", July 1997.
[3] P. Niyogi and M. Belkin, "Laplacian, Eigenmaps and Spectral Techniques for Embedding & Clustering", 2001.
[4] M. Bellkin and P. Niyogig, "Using Manifold Structures of the Partially Labeled Classification" , 2002.
[5] M. Brand, "Charting a Manifold," Neural Info. Processing System, 2002.