ABSTRACT: Machine learning is a field of application in computer science that helps the computers perform tasks by learning the environment, input and output parameters without being explicitly programmed for the task. It is the study and application of algorithms that can learn from observations and make accurate predictions on given datasets – such algorithms can work without having to follow the static, user driven program instructions by making data-driven predictions or self-learning decisions. Face recognition is a biometric method of identifying an individual by comparing live capture or with the digital data of the person stored in the system. It is considered as an application of machine learning. The role of face recognition is growing rapidly to cater Human Computer Interaction (HCI) in various domains. The face recognition problem is addressed in geometric way, which looks at important and distinguishing features, or photometric way, which is a statistical approach that converts an image into values and compares the values with templates to eliminate outliers.

Keywords: Machine Learning, Facial recognition, Matching, Attendance system, Verification, Automatic attendance.

1. INTRODUCTION

Facial recognition activity is a complex activity which depends on various factors like skin colour [16], lighting conditions [18], image source [20], multiple facial expressions [18], angle of capture [19], head count [21] etc. In the proposed work, we have considered only few of the factors namely skin colour [16], lighting conditions [18], head count [21] and image source [20].

The real-world applications are expected to deal with huge data corpus, which results in complex operations in terms of feature extraction and processing. To address some these issues, we are aiming to use one of the dimensionality reduction algorithm called Principal Component Analysis (PCA) [10]. We are intended to make an analysis towards an application automated attendance management using face recognition. The basic function for the face recognition system is to compare the face of a person which is to be recognized with the faces already trained and it has to recognize the best matching face as output even at varied lightening conditions, viewing conditions and facial expressions. Use of face recognition for the purpose of attendance marking is an efficient and time saving application.

1.1 OBJECTIVE

The main objective of this paper is to present a model of an attendance management system using facial recognition which can show the capability of machine learning to the present world. It is aimed to reduce the redundancy and the time consumed by the conventional attendance system. The project uses face recognition through PCA [10] to identify the students in the picture taken by the teacher. The teacher has to just take the picture of the class and store in the database and the system does the rest of the job. The system is also designed to export the processed data into desired reports.
1.3 PROPOSED SYSTEM

In our proposed work, we are planning to draft a system namely automated attendance system using face recognition mechanism in the context of grouped images. The proposed application has a database which stores the pictures of the students taken by the teacher through smart phone. The system then runs an algorithm that extracts the faces using Viola Jones object detection framework as proposed by Viola et al [5] and sends them to the testing database. Then the system does the facial recognition on the faces using training and testing databases and then generates a report that can be used by the stakeholders. The architecture of the system demands modules to be categorised into Face region extraction, Feature extraction, Training and Image extraction.

2. DESIGN AND IMPLEMENTATION:

The real-world applications are expected to deal with huge data corpus, which results in complex operations in terms of feature extraction and processing. To address some of the issues involved, we are aiming to use one of the dimensionality reduction algorithm called Principal Component Analysis (PCA). We are intended to make an analysis towards an application automated attendance management using face recognition. In the proposed work, we have considered only few of the factors namely skin colour, lighting conditions, head count and image source. In our proposed work, we are planning to explore an application of face recognition which is automated attendance. The proposed application has a database which stores the pictures of the students taken by the teacher through smart phone. The system then runs algorithm that extracts the faces as proposed by Jones et al [10] and sends them to the testing database. Then the system does the facial recognition on the faces using training and testing databases and then generates a report that can be used by the stakeholders. The project is designated to have 4 models, to implement the same we have used the following experimental setup.

The project is implemented in four modules:

- Face region extraction Module
- Feature extraction Module
- Training Module
- Image extraction Module

The extraction of the faces plays a major part in the project as the faces if accurately extracted will yield better results in the facial recognition part. For this we have used the Viola-Jones algorithm. Once the faces are extracted the faces are stored in the testing folder that is then used for facial recognition. The working of the algorithm is explained in the figure 2. The Viola–Jones object detection framework proposed by Jones et al [5] is the first object detection framework to provide accurate object detection rates in real-time proposed in 2001 by Paul Viola and Michael Jones. Although it can be trained to detect a variety of objects, it was motivated strongly by the problem of face detection. The proposal suggested by also specifies it to be robust, real time face detection algorithm. The working of the algorithm is explained in the figure.

The input image is accessed to extract Haar Features which are the similar properties that all human faces share. A few properties common to human faces:
- The region around the eyes are darker than the upper cheek region.
- The nose bridge region is brighter than the eye region.

The Haar Features extracted are then dimensionally reduced using the method proposed by Pearson through summed area table [9] which is a data structure and algorithm for quickly generating the sum of values in a rectangular grid. In the image processing domain, it is known as an integral image.

\[
I(x, y) = \sum_{x' \leq x, y' \leq y} I(x', y')
\]

\(x, y\) -- points in the summed area table
\(x', y'\) -- points above or below the input points in the summed area table

The face detection framework by default has a learning algorithm called AdaBoost [11] that can select the best features to train classifiers that use them. This algorithm then constructs a “strong” classifier as a linear combination of simple weak classifiers. Then a cascade of more complex classifiers achieves even accurate detection rates. Cascading strong classifiers are arranged in an order of complexity, where each successive classifier is trained only on those samples which pass through the preceding classifiers.

The Teacher is the user of the system. He/she should take the pictures of the class and then upload them to the system which then processed. The source of image can be anything. The faces after extraction are sent to testing database.
The teacher first takes the pictures of the class and then uploads them to the system which then performs the processing. The faces after extraction are sent to testing database.

**Implementation in MATLAB:**

FDetect1 = vision.CascadeObjectDetector; % To detect the face
FDetect1.MinSize = [50 50]; % To avoid false detections
FDetect1.MergeThreshold = 10; % To avoid false detections

Im = imread('IMG_4388.jpg'); % Read the image
BB1 = step(FDetect1,Im); % Returns the bounding box values based on the number of objects in the image.

```matlab
figure,
imshow(Im); % shows the image
for i = 1: size(BB1,1)
    rectangle ('Position',BB1(i,:), 'LineWidth',3,'LineStyle','-','EdgeColor','r'); % Creates a rectangle
    v=imcrop(I,[BB1(i,1),BB1(i,2),BB1(i,3),BB1(i,4)]);
% Crops the faces
    c=imresize(v,[200 180]); % Resizes the images based on the scale specified
    h='%d.jpg';
    f=sprintf(h,i);
    str1=strcat('G:\Projects\Major Project\matlabcode\a\',f);
    imwrite(c,str1); % Writes the images to the path specified
end;
```

**Implementation in MATLAB:**

```matlab
function [m1, A1, Eigenf] = EigenfaceCore(T1)
% Calculating the mean image
M1 = mean(T1,2); % Computing the average face image
Train1 = size(T1,2);
% Calculating the deviation of each image from mean image
A1= []; % Merging all images
for i = 1 : Train1
    temp1 = double(T1(:,i))-m; % Computing the difference image for each image in training set
    A1= [A1 temp1];
end
for i = 1 : size(V1,2)
    if( D1(i,i)>1 )
        L1eig_vec = [L1eig_vec V1(:,i)];
    end
end
% Calculating the eigenvectors of covariance matrix 'C'.
Eigenf = A1 * L1eigvec; % A: centered image vectors.
```

2.2 FEATURE EXTRACTION MODULE:

In Proposed work the features of the input image are extracted using Principal Component Analysis (PCA) as Fig 4. PCA is one of a dimensionality reduction method which retain majority of the variations present in the data. Steps involved in PCA are:

- **Step 1:** Get data in the input image in form of matrix as A1, A2.... Am.
- **Step 2:** Calculate the mean of the matrices M1 as \( \Sigma A1 / M1 \).
- **Step 3:** Subtract mean from original data A1 - M1.
- **Step 4:** Calculate the covariance matrix C = A1, A2.... Am - Ω.
- **Step 5:** Calculate the eigenvalues of the covariance matrix.

- **Step 6:** Choose components with the highest information and form a feature vector.
- **Step 7:** Derive a new data set and project the Eigen face image.

### Table 1: Training Database

<table>
<thead>
<tr>
<th>Factor</th>
<th>Number of images</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dark complexity</td>
<td>480</td>
<td>180 x 200</td>
</tr>
<tr>
<td>Light complexity</td>
<td>360</td>
<td>180 x 200</td>
</tr>
<tr>
<td>Tilted images (images with angle of capture greater than 45 degrees)</td>
<td>360</td>
<td>180 x 200</td>
</tr>
</tbody>
</table>
The training and testing database are unique in nature. The training database is built by taking care of all the factors we have planned to analyse in the project. In the table above we have documented the various scales and the factors we have taken into account before preparing the database. The database is prepared beforehand and is under the control of the database administrator. The training database has images as per the numbers specified by the table.

**Implementation in MATLAB:**

```matlab
function T = CreateDatabase(TrainDat)
TrainFi = dir(TrainDat);
Train2 = 0;
for i = 1:size(TrainFi,1)
    if
        not(strcmp(TrainFi(i).name,'.')|strcmp(TrainFi(i).name,'..')|strcmp(TrainFi(i).name,'Thumbs.db'))
    Train2 = Train2 + 1; % Number of all images in the training database
    end
end
% Construction of 2D matrix from 1D image vectors
T1 = [];
for i = 1 : Train2
    str = int2str(i);
    str = strcat('\',str,'.jpg');
    str = strcat(TrainDat,str);
    img1 = imread(str);
    [row col] = size(img1);
    temp1 = reshape(img1',row*col,1); % Reshaping 2D images into 1D image vectors
    T1 = [T1 temp1]; % 'T' grows after each turn
End
```

2.4 IMAGE EXTRACTION MODULE

First the input image is sent to the face detection algorithm that detects the faces and extracts them. Then these images are then being sent to PCA algorithm that extracts the features and then compares them with features of the training module and the matches are shown in the results.

**Implementation in MATLAB:**

```matlab
function OutputName = Recognition(TestImage, m, A, Eigenfaces)
ProjImages = [];
TraNumber = size(Eigfaces,2);
for i = 1 : TraNumber
    temp = Eigfaces'*A(:,i);
    ProjImages = [ProjImages temp];
end
InImage = imread(TImage);
[ row col ] = size(temp);
InImage = reshape(temp',row*col,1);
Diff = double(InImage) - m; % Centered test image
ProjTesImage = Eigenf'*Diff; % Test image feature vector
Eucdist = [];
for i = 1 : TraNumber
    q = ProjImages(:,i);
    temp = ( norm( ProjTesImage - q ) )^2;
    Eucdist = [Eucdist temp];
end
[Eucdist_min , Recog_index] = min(Eucdist);
OutName = strcat(int2str(Recog_index),'.jpg');
```

**The Calling function**

```matlab
TrainPath = 'G:\Projects\Major Project\matlabcode\training'; % Training database path
TestDatabasePath = 'G:\Projects\Major Project\matlabcode\testing'; % path of Testing database
pro= { 'Enter test image with number'};
dlgtitle1 = 'Input of PCA-Based Face Recognition System';
umlines1=1;
de1 = [ '1'];
```
TestI = inputdlg(pro,dlgtitle1,numlines1,de1);
TestI = strcat(TestDatabasePath, '\\', char(TestI), '.jpg');
im = imread(TestI);
T1= CreateDatabase(TrainPath); % database creation
[m, A, Eigenfaces] = EigenfaceCore(T); %Calculates eigenfaces,eigenvalues and eigenvectors
OutputName = Recognition (TestImage, m, A, Eigenfaces);
SelectedImage = strcat(TrainDatabasePath, '\\',OutputName);
SelectedImage = imread(SelectedImage);
figure,imshow(SelectedImage); %Shows the matched image
st1 = strcat('The Match is : ',OutputName);
disp (st1);

2.5 SOFTWARE REQUIREMENTS

Programming: MATLAB R2015a
Report Generation: Microsoft Excel
OS: Windows XP and above, Linux- Ubuntu 14 and above, Debian 7.0 and above

2.6 HARDWARE REQUIREMENTS

Processor: Any Intel x86 processor
Hard Disk: 3-4GB for installation
RAM: 2 GB and above
Smartphone or any Camera

3. RESULTS AND OBSERVATIONS

The system is trained by taking account of the only some of the factors that affect face recognition namely skin colour, head count, image source and lighting conditions. The system is trained by the images pertaining to each factor. The testing is also done by taking care the test cases cover the factors that we have taken as our assumptions. Both the testing and training sets are unique in nature. The training database is prepared manually beforehand.

3.1 OUTPUT SCREENS

- Database(Training):

The training database is built by taking care of all the factors we have planned to analyse in the project. The image above shows the images stored in the database. We have documented the various scales and the factors we have taken in account before preparing the database. The database is prepared beforehand and is under the control the database administrator.

- Database(Testing):

The testing database is prepared by extracting the faces in the input image uploaded by the teacher. The images extracted are as per the scale specified in the training database to avoid any discrepancies in the feature extraction.

- Table:

The above table contains the student information along with their attendance details. This table also contains the list of images of the student that are stored in the database. This table is created beforehand and is handled by the database administrator.

- Face Region Extraction:

The above image shows the faces that are detected by the face detection algorithm [5]. The red rectangles cover the faces detected by the system. The algorithm has size of the rectangle and various parameters specified beforehand.
The above image shows the faces that are detected by the face detection algorithm [5]. The faces are extracted and resized as per the scale specified in the training database. The images are extracted and stored in the database as per the path specified in the code.

- Image extraction results:

The image extraction module then compares the images in the training and testing database and then displays the test image and the image it got matched with.

- Generate Excel sheet

The image extraction compares the images and then exports the data to a report which can then be reviewed by the teacher.

3.2 TEST CASES

- Case A: Best case

The above table 2 documents the results as observed by the project in appropriate conditions. The project performs better if the number of faces in the input image is less than 20.

- Case B: Least case

The below table 3 documents the results as observed by the project in inappropriate conditions. The project loses almost 20% of its accuracy if the number of faces in the input image is more than 20.

- Case C: Average case

The below table 4 documents the results as observed by the project in different conditions. The table also documents the detection and recognition rates in various lighting conditions and images sources.
Figure 18: Proposed workflow (Average case)

The below table 5 documents the results as observed by the project in different conditions. The table also documents the detection and recognition rates in various lighting conditions and images sources.

<table>
<thead>
<tr>
<th>Image Source</th>
<th>Light Source</th>
<th>Skin Color</th>
<th>Faces Detected</th>
<th>Faces Recognized (In Percentage)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSLR</td>
<td>Natural</td>
<td>Light</td>
<td>100</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dark</td>
<td>87</td>
<td>78.9%</td>
</tr>
<tr>
<td>Smart-Light</td>
<td>Natural</td>
<td>Light</td>
<td>100</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dark</td>
<td>80</td>
<td>80.8%</td>
</tr>
<tr>
<td>Photos Camera</td>
<td>Natural</td>
<td>Light</td>
<td>100</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dark</td>
<td>89.6</td>
<td>82.3%</td>
</tr>
<tr>
<td>Non-DSLR</td>
<td>Natural</td>
<td>Light</td>
<td>100</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dark</td>
<td>88.4</td>
<td>81.4%</td>
</tr>
<tr>
<td>Smart-Light</td>
<td>Natural</td>
<td>Light</td>
<td>100</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dark</td>
<td>83.5</td>
<td>81.8%</td>
</tr>
</tbody>
</table>

The above table documents the results as observed by the project in different conditions. The table also documents the detection and recognition rates in various lighting conditions and images sources.

The below table 5 documents the results as observed by the project in different conditions. The table also documents the detection and recognition rates in various lighting conditions and images sources.

4. CONCLUSION:

A model of face recognition System PCA and implementation in MATLAB has been discussed. From the results, we can conclude that the proposed system shows better results if the number of faces in the test image faces are not tilted beyond a certain angle. Also, the automated attendance system is a smarter way for marking attendance. The proposed system also exports the data regarding the attendance to an excel sheet thus simplifying the process further for the stakeholders. As such many examples can be derived from the capabilities of machine learning which are driving the present computer generation.

4.1 SCOPE FOR FURTHER ENHACEMENTS:

First the system must be trained must extensively on cases like:
- Increased number of persons in the image
- Presence of faces that have a tilt of 45 degrees and above
- Images having faces with dark complexity
- Use of Artificial Neural networks will definitely increase the accuracy of the system.
- The system can be implemented as a commercial product in educational institutions.
- The system can be further improved by adding dynamic recognition to its foray.

REFERENCES


© 2015-19, IJARCS All Rights Reserved