Invariant Moment Based ID3 Classifier for Face Recognition Satya Sreedevi Redla, Dr. Vamsi Krishna Mangalampalli, Dr. Banitamani Mallik

Satya Sreedevi Redla

Jackson State University, 1400 John R. Lynch Street, Jackson, MS 39217,USA &

Research scholar at Centurion University of Technology & Management, Odisha, India

Dr. Vamsi Krishna Mangalampalli

Chaitanya Institute of Science & Technology, Kakinada, Andhra Pradesh, India

Dr. Banitamani Mallik

Centurion University of Technology & Management, Odisha, India

**Corresponding author E-mail: ssreemaddali@gmail.com*

Abstract- The biometric-based face recognition techniques are considered to be most promising methods in face recognition systems. Many different methods of capturing features of facial image and compare with those images in database have been proposed in last few decades. The challenges in recognition system due to different imaging conditions in real-world like illuminations, expressions, poses and faces with low-resolution made researchers to continue to develop new methods to reduce the huge volumes of data, to increase the efficiency and accuracy of recognition process. This paper develops face recognition methodology based on 7 Hu moments invariant of rotation, translation & scaling and ID3 classifier. Methodology and demonstrations are being provided on Bio id face benchmark data base as proof of concept. It is noticed 91% accuracy is attained on randomly selected sample of 10 individual's faces.

Keywords: Face Recognition, Face detection, Feature extraction, Hu moments, ID3 classifier

INTRODUCTION

In 1960s the first semi-automated system for facial recognition was carried out manually to locate the features like eyes, ears, nose and mouth on the photographs before it calculated ratios and distance to a common reference point. These features were then compared to a reference data base. In 1970 Goldstein and Harmon used 21 specific subjective points such as hair color, lip thickness etc. to automate the face recognition. In 1988, Kirby and Sirovich applied Principal component analysis and a standard linear algebra technique to compress the face images. In 1991, Turk and Pentland [2] discovered that while using eigenfaces technique, the residual errors can be used to detect faces in images for a reliable real time automated face recognition system.

The face recognition system procedure consists of Face Detection, Feature Extraction, and Face Recognition [1].

A work flow of image-based face recognition system proposed by Wei-Lun Chao depicted in the Fig 1.

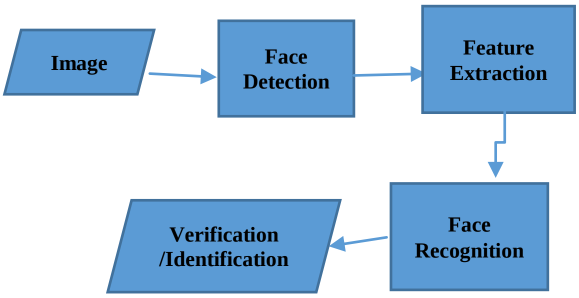


Fig 1: General face recognition system

Face detection is to identify an object as a face in the input image. we first check whether human faces appear in the input image and then find the location of these faces in the image.

After the human-faces are extracted from image. We cannot use directly these faces in recognition system because of certain problems like dimensionality (For example, the dimensionality of each face vector of 512X512 image is 250,000), camera alignments, different facial expressions, illuminations, and may suffer from occlusion and clutter. To overcome these drawbacks, feature extractions are performed. A feature is a piece of information that describes an image or part of an image. Size of the feature and the number of features extracted from a single facial image are significant in system.

After extracting features, w to decide whether the input image is known or unknown based on database of faces.

In Face identification we want the recognition system to tell who it is from a given image; while in face verification, given a face image and a guess of identification, we want the system to tell true or false about the guess.

A questioned image will be pre-processed and a region of interest will be fetched from that image. It is then, submitted to the face recognition system. This paper assumes that the given questioned image as whole is a region of interest. Hence this paper is not considering the pre-processing stage. Also, the questioned image may be in different orientation, which will have an influence on the face recognition.

The present study considers 7 HU moments which were invariant with respect to rotation, translation and scaling [6], [7]. A brief account of this is provided in section II for completeness.

In [7], the Hu central moments of the image function f(x,y)with p as the order of x and q as the of y is defined as

$$\mu_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x-\bar{x})^p(y-\bar{y})^q f(x,y)dxdy$$
$$p,q = 0,1,2,\cdots$$

where

$$\bar{x} = \frac{m_{10}}{m_{00}} \qquad \bar{y} = \frac{m_{01}}{m_{00}} \tag{1}$$

Scale invariance can be obtained by normalization. The normalized central moments are defined as

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}, \ \gamma = (p+q+2)/2, \ p+q = 2,3,\cdots \tag{2}$$

Based on the normalized central moments, Hu introduced seven moment invariants. These seven moment invariants are unchanged under image scaling, translation and rotation are

$$\begin{aligned}
\phi_1 &= \eta_{20} + \eta_{02} \\
\phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
\phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \mu_{03})^2 \\
\phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \mu_{03})^2 \\
\phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\
&\quad + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
\phi_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
&\quad + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
\phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\
&\quad - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[(3\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]
\end{aligned} \tag{3}$$

The following section of this paper provides feature extraction from the given image and the corresponding illustration. Mean, SD, CV and 7 invariant moments are considered as features and be computed for the given images. The individual person id of the image is labeled as class. The Data is divided into two parts. One will be training part and the other will be testing part. Data preparation is explained in section II along with statistics.

In this paper a ID3 classifier has been built in Matlab by using training data set. The performance of the model is tested on the test data set.

Bio id face benchmark data base is considered for the demonstration of the proposed methodology. Rest of the paper as follows

2. DATA PREPARATION & PROPOSED METHODOLOGY

Ten features namely Mean, SD, CV and 7 invariant moments for each image were computed as a real vector of size 10. A class label is the person id of the image and it is a categorical data type. Thus each image is represented by a vector of 10 real numbers with one categorical data item. Fig 2 is the given image for which the features are computed and represented as vector as shown below.

.

Fig2: A Given image (S_id 180)

Mean = 107.5078 IM3= 2.39E-11

SD= 67.10222 IM4=1.41E-11

CV= 62.41611	IM5=1.63E-22
IM1 =0.001377	IM6=-6E-15
IM2= 1.84E-07	IM7= 3.91E-23

Bio Id Face Database: This dataset consists of 1521 gray level images with a resolution of 384x286 pixel of different individuals with different frequencies. The Database was recorded and published for the researchers to work in the area of face detection and to compare the quality of their algorithms with others. During the recording the images a special emphasis has been laid on "real world" conditions. Therefore, the test set features a large variety of illumination, background and face size. For sample images please visit the BioID Face DB page.

Images pertaining to 10 individuals with reasonable frequency has been selected for demonstration purpose. Hence the data set consist of 445 images of 10 individuals with ids 180, 260, 322, 418, 449, 483, 519, 660, 741, and 973. A sample image of each of 10 individuals is given in the following Fig 3 and their labelled ids in table I

Table I: Selected Persons Ids (S_id)

S_id:	180	260	322	418	449	483	519	660	741	973

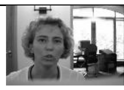
180	260	322	418	449
				
483	519	660	741	973
				

Fig 3:Sample faces of Selected 10 individuals corresponding to Table I

In Table II, ISNo represents the image id, S id represents the person id, mean, SD, CV, IM1, IM2, IM3, IM$, IM5, IM6 and IM7 are the features of these 10 individuals and Class is nothing but S_id.

The following provides a sample of 6 images of an individual labelled as 180 as shown in Fig 4.

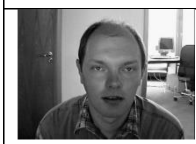
1	2
	
3	4
	
5	6

Fig 4: A sample six faces of person S_id =180

The Box plots of the 6 image in Fig4 is given in Fig 5 respectively. These Box plots displays range and distribution of data along the number line.

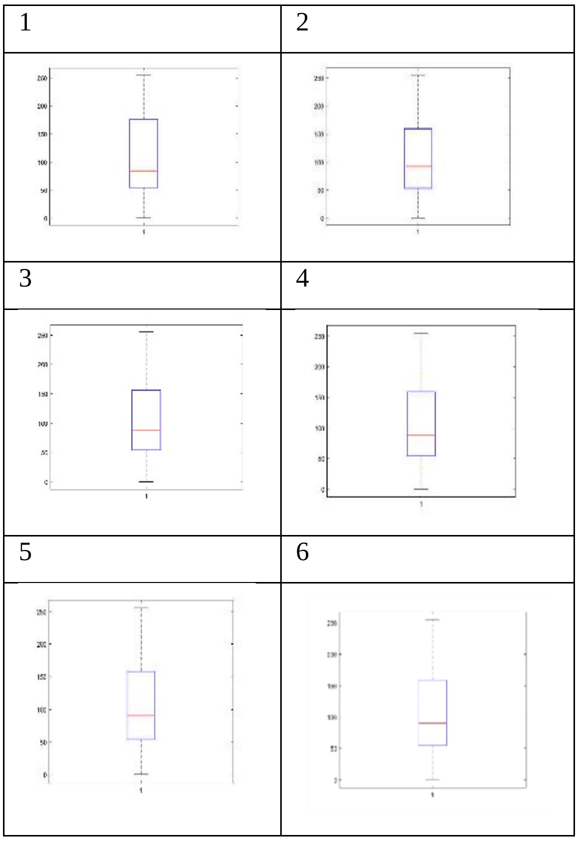


Fig 5: Box Plots corresponds the image data given in fig4

Analysing the box plots, one can notice variation in the plots corresponds to 6 images of the same individual. This challenges the face recognition system.

The composition of 445 images in our demonstrative data base related to 10 individuals is in Table III

S_id	Number of Images
180	42
260	48
322	92
418	31

449	33
483	36
519	37
660	46
741	38
973	42

Table III: Representation of faces of the selected 10 individuals in the demonstrative Data set of size 445

The abstract aspect of the features of the 445 images is given in the Table IV and Table V. The average of the 10 features listed in table IV and Standard deviation of the 10 features in Table V respectively.

Histogram of all the 10 features of the selected data set is provided in Fig

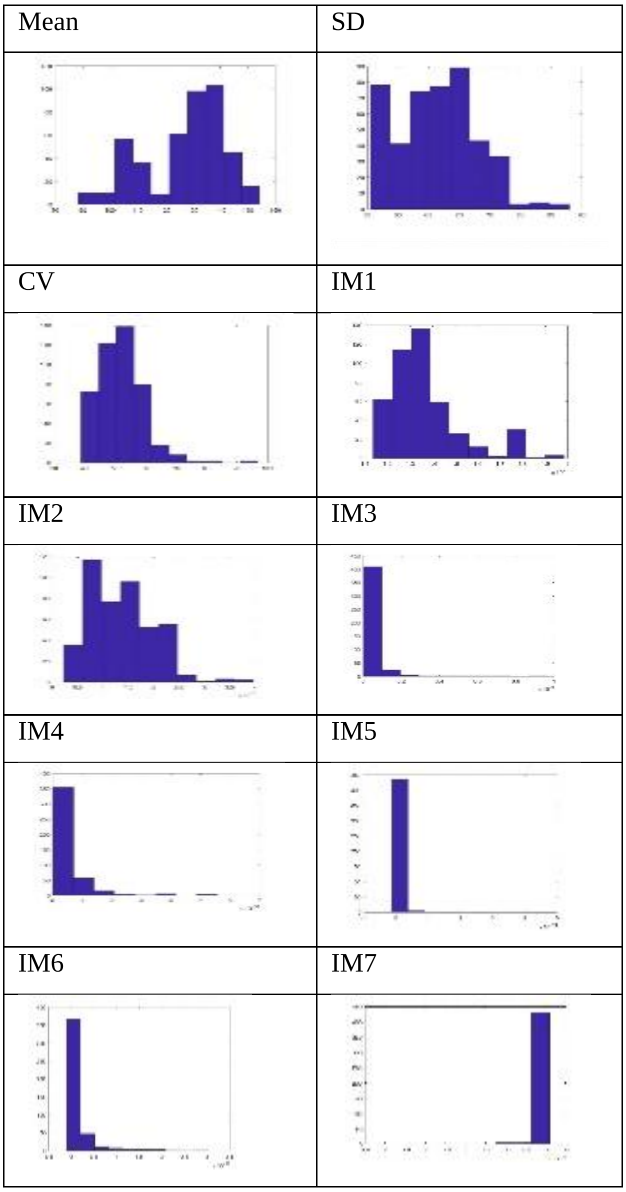


Fig 6: Histograms of the Features of Data Set

3. PERFORMANCE OF PROPOSED METHODOLOGY IVMID3

The given images are divided into Training and Testing Image Sets. The considered 445 Images of Bio Id face bench mark data set are divided into training and testing data sets with sizes 299 and 146 respectively by maintaining proper representation of each individual.

The corresponding features are extracted and instructed as training and testing data.

Training data is used to train the ID3 classifier

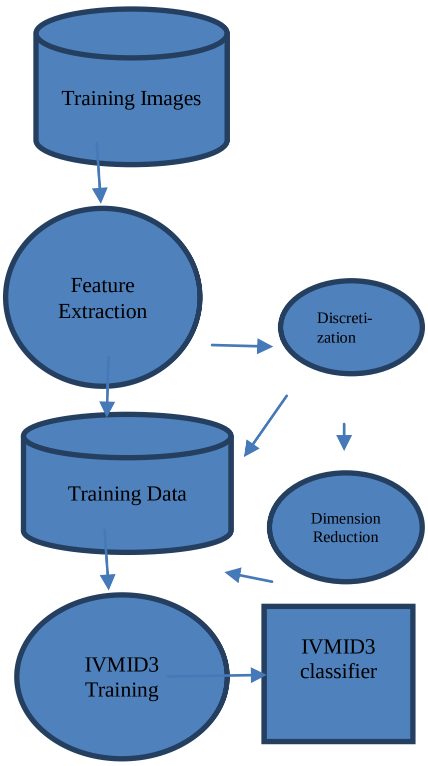
Testing data is used for validating trained classifier by training set.

In this paper the features are invariant moments and the classifier is ID3. Hence the proposed classifier named as in short IVMID3.

The following fig 7, 8 & 9 explains the building process of proposed methodology IVMID3.

Building a IVMID3 classifier using the training data set of images is given in fig7

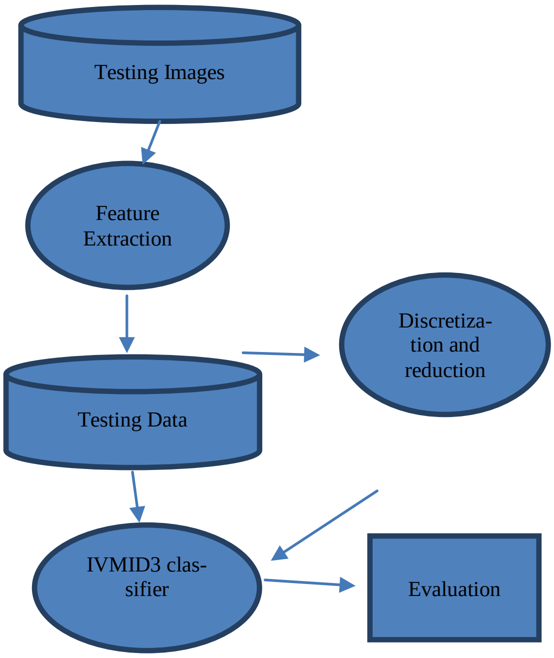
Fig 7: Training IVMID3



The validation steps of the IVMID3 is given in flow chart below.

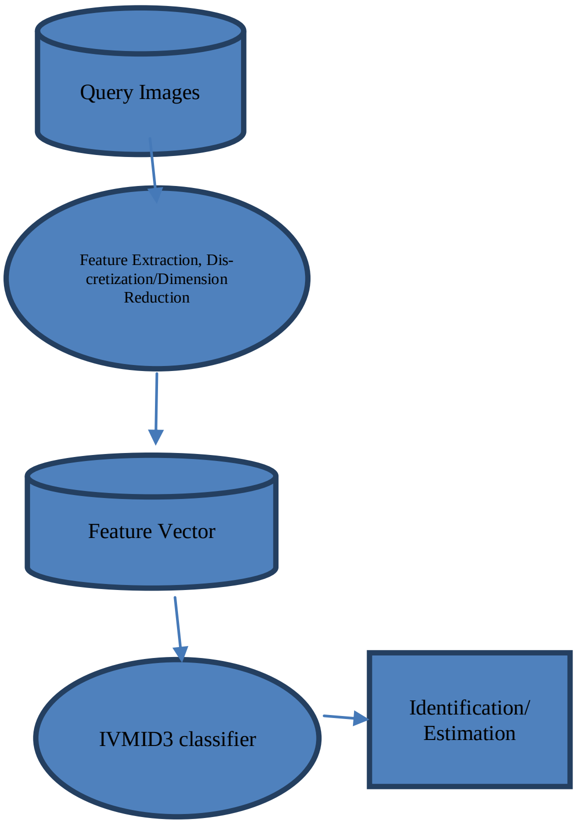
Testing data is used for validating trained classifier IVMID3

Fig 8: Evaluation/Validation IVMID3



Face Recognition of a questioned image can be obtained by following steps as given in the following flow chart.

Fig 9: Identification/Estimation IVMID3



The Mean, Standard Deviation, CV of 10 features from 455 images are computed and listed in the following Table VI.

Table VI: Mean, SD, CV of features

	Mean	SD	CV	IM1	IM2	IM3	IM4	IM5	IM6	IM7	class
Mean	127.3593933	66.2959909	52.67542088	0.0013606	1.36791E-07	5.65165E-11	5.50912E-11	3.92157E-21	1.44788E-14	-5.21998E-21	479.5820225
Std	14.60524582	6.541528751	7.596274292	0.000163257	6.38067E-08	8.08688E-11	7.70735E-11	3.11547E-20	3.14213E-14	3.25518E-20	227.0767236
CV	11.46774136	9.86715586	14.42090858	11.99886664	46.64537158	143.0888439	139.9016765	794.444928	217.0154216	-623.6006473	47.34888152

After extracting the selected features of the images, the following discretization technique is used.

Lower Integer part of ((x-mean)/σ).k)+KK is used to discretization of the data of each image.

With k 7

KK 28

Using above discretization factor a Discretized data of sample data of 10 images is given in Table VII

Table VII: Discretized data

D_Mean	D_SD	D_CV	D_IM1	D_IM2	D_IM3	D_IM4	D_IM5	D_IM6	D_IM7	Class
18	28	36	28	33	25	24	27	23	29	180
21	26	32	30	19	29	36	30	31	29	260
40	22	15	18	24	25	26	27	27	28	322
33	35	28	24	27	23	26	27	24	29	418
31	34	28	25	27	23	27	26	26	29	449
35	44	32	18	16	25	27	26	26	28	483
29	22	22	27	28	26	23	27	25	29	519
25	22	25	35	38	28	23	27	24	29	660
31	17	17	21	20	28	27	27	27	28	741
34	28	22	21	33	30	24	27	26	29	973

Where D- Stands for Discretization, IM- Invariant Moment

Rough sets theory provides a novel approach to knowledge description and to approximation of sets. In rough sets theory, feature values of sample objects are collected in what are known as information tables. Rows of such a table correspond to objects and columns correspond to object features. Rough set theory provides a mathematical tool that can be used for both feature selection and knowledge discovery. It helps us to find out the minimal attribute sets called ‘Reduct’ to classify objects without deterioration of classification quality

Table VIII: Reduct computation

Reduct	Mean	SD	IMV6	IMV2	IMV4
Kapas	0.193258427	0.948314607	0.993258427	0.995505618	1
A	445	359	23	3	2
Time	0.096713079				

The Reduct is Mean, SD, IM6, IM2 and IM4

Reduced discretization Table IX of 10 images is as follows

Table IX: Reduct of discretized data

Mean	SD	IMV6	IMV2	IMV4	Class
18	28	23	33	24	180
21	26	31	19	36	260
40	22	27	24	26	322
33	35	24	27	26	418
31	34	26	27	27	449
35	44	26	16	27	483
29	22	25	28	23	519
25	22	24	38	23	660
31	17	27	20	27	741
34	28	26	33	24	973

ID3 algorithm invented by Ross Quinlan [8] which is used to generate a decision tree from a dataset. Using Matlab, the code of the following algorithm ID3 decision tree has been built

The ID3 decision tree for the Bio id face bench mark data is as follows

```
% id3tree for data set =Bio Sampled Reduct

function classify = f_id3tree_BioSample_Reduct(data)
D_mean = data(1);
switch D_mean
   case 9
      classify = 322;
   case 10
      classify = 322;
   case 11
      classify = 322;
   case 13
      classify = 322;
   case 14
      classify = 322;
   case 15
      classify = 660;
   case 16
      D_std = data(2);
      switch D_std
         case 17
            classify = 660;
         case 20
```

```
            classify = 322;
        otherwise  classify = 999; %default case
    end
case 17
    classify = 322;
case 18
    D_std = data(2);
    switch D_std
        case 23
            classify = 322;
        case 24
            classify = 322;
        case 25
            classify = 322;
        case 27
            classify = 180;
        case 28
            classify = 180;
        case 29
            classify = 180;
        case 31
            classify = 180;
        otherwise  classify = 999; %default case
    end
case 19
    D_std = data(2);
    switch D_std
        case 24
            classify = 180;
        case 25
            classify = 260;
        case 29
            classify = 180;
        case 30
            classify = 180;
        case 31
            classify = 180;
        case 36
            classify = 322;
        otherwise  classify = 999; %default case
    end
case 20
```

```
        D_std = data(2);
        switch D_std
            case 24
                classify = 260;
            case 25
                classify = 260;
            case 26
                classify = 260;
            case 36
                classify = 322;
            otherwise  classify = 999; %default case
        end
    case 21
        D_std = data(2);
        switch D_std
            case 26
                classify = 260;
            case 42
                classify = 322;
            otherwise  classify = 999; %default case
        end
    case 22
        classify = 322;
    case 23
        D_std = data(2);
        switch D_std
            case 17
                classify = 260;
            case 28
                classify = 973;
            case 41
                classify = 322;
            case 47
                classify = 322;
            otherwise  classify = 999; %default case
        end
    case 24
        classify = 973;
    case 25
        D_std = data(2);
        switch D_std
            case 22
```

```
            classify = 660;
         case 23
            classify = 660;
         case 29
            classify = 973;
         case 30
            classify = 973;
         case 31
            classify = 973;
         otherwise  classify = 999; %default case
      end
   case 26
      D_std = data(2);
      switch D_std
         case 22
            classify = 660;
         case 23
            classify = 660;
         case 30
            classify = 973;
         case 31
            classify = 973;
         case 32
            classify = 973;
         otherwise  classify = 999; %default case
      end
   case 27
      D_std = data(2);
      switch D_std
         case 21
            classify = 260;
         case 22
            classify = 260;
         case 29
            classify = 973;
         case 32
            classify = 973;
         otherwise  classify = 999; %default case
      end
   case 28
      classify = 260;
   case 29
```

```
        D_std = data(2);
        switch D_std
            case 16
                classify = 741;
            case 17
                classify = 741;
            case 23
                classify = 519;
            case 24
                classify = 519;
            otherwise  classify = 999; %default case
        end
    case 30
        D_std = data(2);
        switch D_std
            case 17
                classify = 741;
            case 18
                classify = 741;
            case 24
                classify = 519;
            case 25
                classify = 519;
            case 35
                classify = 322;
            case 36
                classify = 322;
            otherwise  classify = 999; %default case
        end
    case 31
        D_IM2 = data(4);
        switch D_IM2
            case 19
                classify = 741;
            case 20
                classify = 741;
            case 25
                classify = 519;
            case 26
                classify = 483;
            case 27
                classify = 449;
```

```
        case 28
          classify = 449;
        case 29
          classify = 322;
        case 30
          classify = 418;
        case 32
          classify = 418;
        otherwise  classify = 999; %default case
      end
    case 32
      D_std = data(2);
      switch D_std
        case 17
          classify = 260;
        case 18
          D_IM6 = data(3);
          switch D_IM6
            case 25
              classify = 260;
            case 26
              classify = 741;
            otherwise  classify = 999; %default case
          end
        case 19
          classify = 260;
        case 20
          classify = 260;
        case 28
          classify = 322;
        case 29
          classify = 418;
        case 30
          classify = 418;
        case 31
          classify = 418;
        case 32
          classify = 449;
        case 33
          D_IM6 = data(3);
          switch D_IM6
            case 24
```

```
                D_IM2 = data(4);
                switch D_IM2
                    case 27
                        classify = 449;
                    case 28
                        classify = 418; %majority voting
                    otherwise  classify = 999; %default case
                end
            case 25
                classify = 449;
            case 26
                classify = 418;
            otherwise  classify = 999; %default case
        end
    case 36
        classify = 322;
    case 37
        classify = 322;
    otherwise  classify = 999; %default case
end
case 33
    D_std = data(2);
    switch D_std
        case 18
            classify = 260;
        case 19
            classify = 260;
        case 28
            classify = 322;
        case 29
            classify = 322;
        case 30
            classify = 418;
        case 31
            D_IM6 = data(3);
            switch D_IM6
                case 24
                    classify = 418;
                case 25
                    classify = 418;
                case 26
                    classify = 418;
```

```
                case 30
                    classify = 322;
                otherwise  classify = 999; %default case
            end
        case 32
            classify = 418;
        case 33
            classify = 322;
        otherwise  classify = 999; %default case
    end
case 34
    D_std = data(2);
    switch D_std
        case 19
            classify = 260;
        case 28
            classify = 973;
        case 29
            classify = 322;
        case 30
            classify = 449;
        case 32
            classify = 418;
        case 33
            classify = 418;
        case 35
            D_IM6 = data(3);
            switch D_IM6
                case 24
                    classify = 449;
                case 25
                    classify = 449;
                case 26
                    classify = 322;
                otherwise  classify = 999; %default case
            end
        case 36
            classify = 322;
        case 37
            classify = 483;
        case 38
            classify = 483;
```

```
        case 46
           classify = 483;
        case 48
           classify = 483;
        otherwise  classify = 999; %default case
     end
  case 35
     classify = 483;
  case 36
     classify = 483;
  case 37
     classify = 322;
  case 38
     classify = 973;
  case 39
     classify = 322;
  case 40
     classify = 322;
  otherwise  classify = 999; %default case
end
```

A 10-fold validation on the reduced discretised data is given in below Table X.

Tables: X On Reduced Discretized

fold	1	2	3	4	5	6	7	8	9	10	mean	SD	CV
ID3	91.11	88.89	95.56	97.78	88.89	95.56	95.56	97.78	95.56	92.50	93.92	3.35	3.57

4. CONCLUSIONS

A same ID3 builder is also carried on discretized data without dimension reduction. The performance is very low for this bench mark data   Whereas Reduced data is giving significantly high performance on this bench mark data around 91% This study reveals with 5 dimensional features   are sufficient for face recognition system.

REFERENCES

[1] Wei-Lun Chao, “Face Recognition”, GICE, National Taiwan University

[2] M. A. Turk and A. P. Pentland, "Face Recognition Using Eigenfaces", 1991.

[3]R. Brunelli and T. Poggio, \Face recognition: features versus templates," IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 15, October 1993

[4] S. Suhas, A. Kurhe, Dr. P. Khanale, “Face Recognition Using Principal Component Analysis and Linear Discriminant Analysis on Holistic Approach in Facial Images Database”, IOSR Journal of Engineering e-ISSN: 2250-3021, p-ISSN: 2278-8719, Vol. 2, Issue 12 (Dec. 2012), ||V4|| PP 15-2

[5] Divyarajsinh N. Parmar, Brijesh B. Mehta “Face Recognition Methods & Applications”, Int. J. Computer Technology & Applications, Vol 4 (1), 84-8

[6] MING-KUEI HU,” Visual Pattern Recognition by Moment Invariants”, IRE TRANSACTIONS ON INFORMATION THEORY, 1962.

[7] Z Huang, J Leng, “Analysis of Hu's moment invariants on image scaling and rotation”, Computer Engineering and Technology …, 2010 - ieeexplore.ieee.org [Volume 7]

[8] Quinlan, J. R. 1986. Induction of Decision Trees. Mach. Learn. 1, 1 (Mar. 1986), 81–106

Table II: Features of the images given in Fig 3

IS No	S_id	NR	NC	min	max	mean	SD	CV	IM1	IM2	IM3	IM4	IM5	IM6	IM7	class
1	180	286	384	0	255	107.5078	67.10222	62.41611	0.001377	1.84E-07	2.39E-11	1.41E-11	1.63E-22	-6E-15	3.91E-23	180
43	260	286	384	0	255	113.6477	64.99283	57.18797	0.001417	5.65E-08	7.49E-11	1.44E-10	1.45E-20	3.11E-14	8.53E-22	260
91	322	286	384	27	255	154.3289	60.80314	39.39841	0.001142	1.08E-07	2.82E-11	3.47E-11	2.71E-22	1.01E-14	-1.2E-21	322
183	418	286	384	0	255	138.5516	73.16693	52.80844	0.00127	1.34E-07	3.01E-12	4.07E-11	-4.1E-22	-2.6E-15	1.77E-21	418
214	449	286	384	0	255	135.5407	72.46299	53.46215	0.001298	1.29E-07	4.37E-12	4.65E-11	-2E-21	6.39E-15	1.37E-21	449
247	483	286	384	0	255	143.214	82.13398	57.35054	0.00114	3.2E-08	2.2E-11	5.43E-11	-2.8E-21	7.96E-15	-8.6E-22	483
283	519	286	384	0	255	131.3982	61.39059	46.72101	0.001359	1.4E-07	3.51E-11	8.39E-12	-4.4E-23	2.88E-15	-3.8E-23	519
320	660	286	384	0	255	122.423	61.04817	49.86656	0.001533	2.36E-07	6.64E-11	4.25E-12	-5.5E-24	6.13E-16	1.79E-23	660
366	741	286	384	11	255	134.4783	56.12519	41.7355	0.001198	7.04E-08	5.71E-11	4.79E-11	1.06E-21	1.13E-14	-2.5E-21	741
404	973	286	384	0	255	141.8279	66.46154	46.8607	0.001216	1.83E-07	8.04E-11	1.65E-11	4.34E-22	6.68E-15	-1.9E-22	973

Table IV: Averages of the Features of the images of the considered Data Set

S_id	#	Average of mean	Average of SD	Average of cv	Average of IM1	Average of IM2	Average of IM3	Average of IM4	Average of IM5	Average of IM6	Average of IM7
180	42	107.8923299	66.68177405	61.80275916	0.001406273	1.62381E-07	7.05213E-11	6.6784E-11	-2.99866E-21	8.76015E-15	-3.29988E-21
260	48	126.0425167	61.07764908	49.1227703	0.001300714	6.44151E-08	5.32655E-11	6.78438E-11	5.02982E-21	1.39462E-14	2.15414E-21
322	92	127.4163091	70.0131882	56.45131684	0.001390854	1.6056E-07	1.00354E-10	1.26747E-10	1.82485E-20	4.72295E-14	-2.3999E-20
418	31	138.1173217	69.7933661	50.53478523	0.001313717	1.58056E-07	1.13629E-11	3.49127E-11	-9.8822E-22	1.75065E-15	4.52984E-22
449	33	139.9084734	69.98050072	50.03894005	0.00124647	8.87394E-08	1.55169E-11	2.31632E-11	-4.17611E-22	2.00903E-15	4.00224E-22
483	36	142.5492677	76.92565848	53.98176995	0.001266475	7.64848E-08	2.23992E-11	4.5745E-11	-1.86136E-21	7.925E-15	-6.19209E-22
519	37	132.9486781	63.0344515	47.41231889	0.001331525	1.16865E-07	3.5038E-11	6.98535E-12	-4.81473E-23	1.977E-15	-3.73271E-23
660	46	112.0124274	58.67501109	52.62362422	0.001654063	2.33206E-07	7.10076E-11	1.00689E-11	1.93599E-22	3.89548E-15	2.69373E-22
741	38	132.4420561	56.62398146	42.75627727	0.00121522	6.82216E-08	6.07489E-11	5.03811E-11	1.69671E-21	1.20425E-14	-2.54219E-21
973	42	124.6726149	69.11499711	55.53676556	0.001357795	1.89596E-07	4.42072E-11	1.58041E-11	-2.24356E-22	-1.31438E-15	2.02166E-23
(blank)											
Grand Total	445	127.3593933	66.2959909	52.67542088	0.0013606	1.36791E-07	5.65165E-11	5.50912E-11	3.92157E-21	1.44788E-14	-5.21998E-21

Table V: Standard deviation of the Features of the images of the considered Data Set

S_id	#	SD of mean	SD of SD	SD of CV	SD of IM1	SD of IM2	SD of IM3	SD of IM4	SD of IM5	SD of IM6	SD of IM7
180	42	0.824533435	1.150398815	0.864460715	2.80865E-05	2.11973E-08	2.83865E-11	2.52226E-11	1.47033E-21	8.49118E-15	3.73173E-21
260	48	12.14403886	3.046182802	7.127712865	0.00012223	2.02334E-08	2.39815E-11	4.83394E-11	5.40814E-21	1.11935E-14	3.3284E-21
322	92	19.60362721	6.044078985	11.16742099	0.000214785	6.57497E-08	1.63525E-10	1.37027E-10	6.66102E-20	5.68488E-14	6.85424E-20
418	31	1.614422904	1.409036464	1.001779154	2.39539E-05	1.50694E-08	5.45695E-12	6.46398E-12	6.00396E-22	3.27828E-15	5.2999E-22
449	33	2.151265997	1.75029161	1.752143468	3.76088E-05	2.83473E-08	6.76047E-12	8.67483E-12	3.29444E-22	1.25919E-15	4.95687E-22
483	36	1.679530631	2.302396832	2.065828019	4.66729E-05	1.71783E-08	7.1987E-12	8.42904E-12	7.49398E-22	2.04678E-15	3.28875E-22
519	37	0.709928523	0.498878652	0.218464273	1.10895E-05	8.02267E-09	2.95093E-12	3.60617E-12	3.59379E-23	1.34861E-15	3.00704E-23
660	46	10.48716471	2.576502386	2.652843016	0.000119239	1.16519E-08	1.40471E-11	8.74462E-12	2.91676E-22	3.82551E-15	3.32096E-22
741	38	1.729263129	0.697334405	0.42905652	2.46653E-05	3.19654E-09	1.13057E-11	6.02517E-12	8.91562E-22	1.98721E-15	4.70879E-22
973	42	4.966164976	1.420885015	2.674078596	5.81373E-05	2.91341E-08	1.72529E-11	6.35678E-12	2.70874E-22	3.91712E-15	4.04054E-22
Grand Total	445	14.60524583	6.541528751	7.596274292	0.000163257	6.38067E-08	8.08687E-11	7.70735E-11	3.11547E-20	3.14213E-14	3.25518E-20