Comparing Human Faces using Edge Weighted Dissimilarity Measure

Aditya Nigam
ICARCV 2010, Singapore

Ph.D Student
Department of Computer Science and Engineering
Indian Institute of Technology Kanpur

December 10, 2010

Table of contents

1. Problem Definition
   - Statement and Motivation
2. Related Work
   - *HD* and its Variants
3. Proposed *EWDM* Measure
   - Pre-Processing
   - Defining *EWDM* Measure
4. Experimental Results and Analysis
   - Results
5. Conclusion
6. Efficient Computation of *EWDM*
   - Algorithm
   - Analysis

Problem Definition

- Face picture acquisition under the same physical conditions is not always possible.
- Different face recognition algorithms perform poorly in typical varying environments.
- Varying illumination, poses, lighting conditions, expressions, backgrounds, scales causes a lot of variation in pixels intensities, and hence different algorithms performance got severely affected.
- So we require an algorithm that is robust enough to small amount of such variations.

Motivation

- Edge images are less affected by illumination variations, but they don't carry overall facial appearance "*they contains primarily the structure of the faces*".
- Gray images can't be used directly as they are affected by this illumination variation.
- *EWDM* measure can compare the gray images and is found to be robust to slight variation in pose, expression and illumination.

Hausdorff Distance (*HD*)

- Conventional Hausdorff distance is dissimilarity between two set of points.
- Let $A = \{a_1, a_2, a_3, a_4..a_m\}$ and $B = \{b_1, b_2, b_3, b_4..b_n\}$ be two Set of points then, undirected Hausdorff distance [8] between A and B is defined as:

$$HD(A, B) = HD(B, A) = max(hd(A, B), hd(B, A))$$

here *hd(A,B)* is the directed Hausdorff distance defined by:

Directed hd

$$hd(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\|$$

and, $\|.\|$ is the norm of the vector.

HD Example

<table>
<tr><th>Pairs of Points</th><th>Distances</th><th>Min Value and Correspondance</th><th>Max Value</th></tr>
<tr><td>1-a</td><td>10</td><td rowspan="2">10(1-a)
1 corresponds to a</td><td rowspan="6">12(3-a)
This is the worst correspondance
[Most Dissimilar Points]</td></tr>
<tr><td>1-b</td><td>14</td></tr>
<tr><td>2-a</td><td>8</td><td rowspan="2">8(2-a)
2 corresponds to a</td></tr>
<tr><td>2-b</td><td>10</td></tr>
<tr><td>3-a</td><td>12</td><td rowspan="2">12(3-a)
3 corresponds to a</td></tr>
<tr><td>3-b</td><td>15</td></tr>
</table>

Figure: Example hd(A,B)

PHD

- *HD* measure does not work well when some part of the object is occluded or missing.
- For partial matching partial Hausdorff distance *PHD* was introduced.
- Undirected *PHD* is defined as:

$$PHD(A, B) = PHD(B, A) = max(phd(A, B), phd(B, A))$$

here *phd(A,B)* is the directed *PHD*, which is defined by:

Directed phd

$$phd(A, B) = K^{th} \max_{a \in A} \min_{b \in B} \|a - b\|$$

- Both *HD* and *PHD* works on edge map and can tolerate small amount of local and non-rigid distortion.

MHD

- *MHD* [15] has been introduced that uses averaging which is a linear function which makes it less sensitive to noise.
- Undirected *MHD* is defined as:

$$MHD(A,B) = MHD(B,A) = max(mhd(A,B), mhd(B,A))$$

here *mhd(A,B)* is the directed *MHD*, which is defined by:

Directed mhd

$$mhd(A,B) = \frac{1}{N_a} \sum_{a \in A} \min_{b \in B} \|a - b\|$$

Where N_a is the number of points in set A.

M2HD

- *MHD* is improved to *M2HD* [10] by adding 3 more parameters :

Parameters

Neighborhood function ($\boldsymbol{N_B^a}$) N'hood of the point a in set B

Indicator variable (I) $I = 1$ if a's corresponding point lie in N_B^a else $I = 0$

Associated penalty (P) if $I = 0$ penalize with this penalty

and directed *M2HD* is defined as:

Directed m2hd

$$m2hd(A, B) = \frac{1}{N_a} \sum_{a \in A} d(a, B)$$

Where *d(a,B)* is defined as:

$$d(a, B) = \max[(I \cdot \min_{b \in N_B^a} \|a - b\|), ((1 - I) \cdot P)]$$

SWHD and *SW2HD*

- For better discriminative power *HD* and *M2HD* measures were improved by assigning the weights to every point according to its spatial information.
- Crucial facial feature points like eyes and mouth are approximated by the rectangular windows and are given more importance than others.
- Directed *SWHD* and *SW2HD* [11] were defined as:

Directed swhd and sw2hd

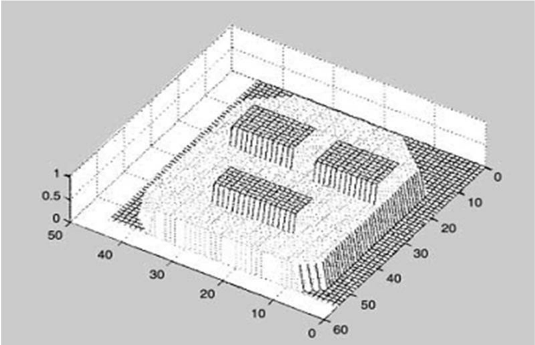
$$\begin{aligned} swhd(A,B) &= \max_{a \in A}\left[w(a) \cdot \min_{b \in B} \|a-b\|\right] \\ sw2hd(A,B) &= \frac{1}{N_a}\sum_{a \in N_a}\left[w(a) \cdot \min_{b \in B} \|a-b\|\right] \end{aligned}$$

Spatial Weighing Function

Where $w(x)$ is defined as:

Weighing Function

$$w(x) = \begin{cases} 1 & x \in \text{Important facial region} \\ W & x \in \text{Unimportant facial region} \\ 0 & x \in \text{Background region} \end{cases}$$



SEWHD and *SEW2HD*

- Rough estimation of facial features cannot fully reflect the exact structure of human face.
- Regions where the difference among the training images is large, the corresponding regions at the eigenfaces will have large magnitude.
- Eigenfaces appears as light and dark areas arranged in a specific pattern. Regions where the difference among the training images is large, the corresponding regions in the eigenfaces will have large magnitude.

Eigen Weighing

Eigen faces can be used as weighing function because they represents the most significant variations in the set of training face images.

Eigen Faces

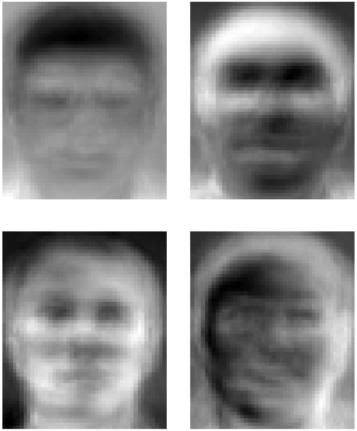


Figure: Eigenfaces

Defining *SEWHD* and *SEW2HD*

- Proposed *SEWHD* and *SEW2HD* [12] are defined as:

Directed sewhd and sew2hd

$$
\begin{aligned}
sewhd(A,B) &= \max_{a \in A} \left[w_e(a) \cdot \min_{b \in B} \|a - b\| \right] \\
sew2hd(A,B) &= \frac{1}{N_a} \sum_{a \in N_a} \left[w_e(a) \cdot \min_{b \in B} \|a - b\| \right]
\end{aligned}
$$

where $w_e(x)$ is defined as:

$w_e(x) =$ Eigen weight function generated by the first eigen vector

EWDM Measure

EWDM

- *EWDM* measure can be applied on *gt*-transformed images obtained from gray-scale facial images.
- *EWDM* measure is similar to the *HD* based measures but is computationally less expensive and more accurate.
- *EWDM* also shows robustness against slight variation in pose, expression and illumination.

Transformation

- A pixel's relative gray value in its neighborhood can be more stable than its own gray value.
- *SK*-transformation [14] provides some robustness against illumination variation and local non-rigid distortions by converting gray scale images into transformed images that preserve intensity distribution.
- Every pixel is represented by an 8-element vector which in itself can store the sign of first-order derivative with respect to its 8-neighborhood.

Property of *SK-transformed images*

Gray value of pixels are being changed in different poses of the same subject but their corresponding vector do not change by a great extent.

Example

Gray values

12	122	26
170	170	178
82	220	211

Sign of first-order derivative

-1	-1	-1
0	X	1
-1	1	1

0	-1	-1	-1	1	1	1	-1

Transformed vector

Problem

- The above property holds when gray values of neighborhood pixels are not too close to each other.
- Usually, we have small variations in the gray values (*e.g.* in background, facial features *etc.*), where the above property *fails* to hold.

Observation

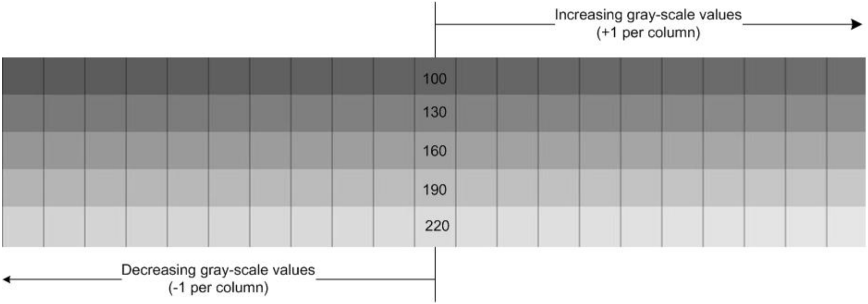


Figure: Gray-value spectrum.

Gray levels are hardly distinguishable (Similar) within a range of ± 5 units.

Improvement

- Basic Comparator

$$X \begin{cases} = X \\ < \alpha \in (X, 255] \\ > \alpha \in [0, X) \end{cases}$$

Improvement

- Basic Comparator

$$X \begin{cases} = X \\ < \alpha \in (X, 255] \\ > \alpha \in [0, X) \end{cases}$$

- *gt*-Comparator

$$X \begin{cases} = \alpha \in [(X - gt), (X + gt)] \\ < \alpha \in (X + gt, 255] \\ > \alpha \in [0, X - gt) \end{cases}$$

Improvement

- Basic Comparator

$$X \begin{cases} = X \\ < \alpha \in (X, 255] \\ > \alpha \in [0, X) \end{cases}$$

- *gt*-Comparator

$$X \begin{cases} = \alpha \in [(X - gt), (X + gt)] \\ < \alpha \in (X + gt, 255] \\ > \alpha \in [0, X - gt) \end{cases}$$

Where

- gt is gray value tolerance, $gt \geq 0$.
- X is a gray level *not merely a number*.
- Gray level X is neither greater than gray level $(X - 1)$ nor less than gray level $(X + 1)$; ideally they should be considered as similar.

Diagrammatically

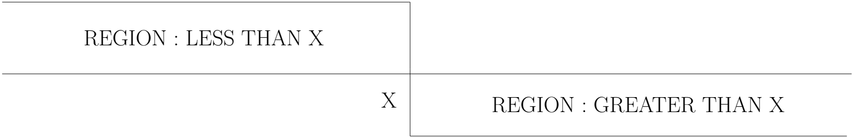


Figure: Basic Comparator

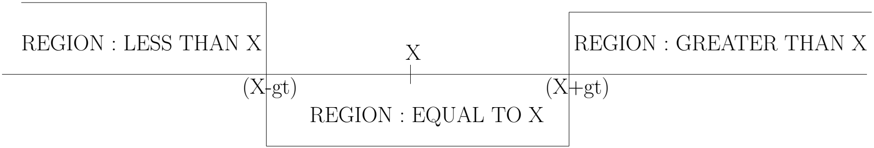


Figure: *gt*-Comparator

gt-Transformation

- Any pixel 'a' is represented by an 8-element vector $V(a)$ whose elements are drawn from the set $\{0, 1, 2\}$.
- The decimal equivalent of the $V(a)$ is called the transformed value of the pixel a, ranging from 0 to 6560 ($= 3^8 - 1$).

Stability

In typical varying environment transformed value of a pixel remains more stable than its corresponding gray value.

gt-Transformed Images

Encoding

Less Than $<$ *RED i.e.*[0], Equal To $=$ *BLUE i.e* [1], Greater Than $>$ *GREEN i.e.* [2].

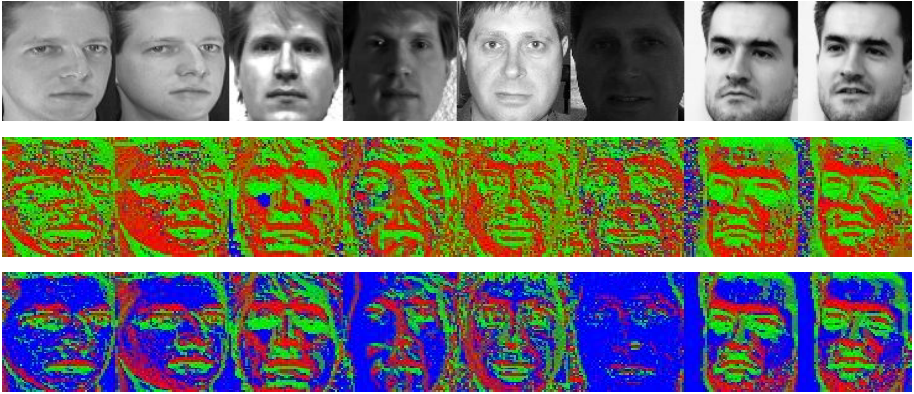


Figure: gt-Transformed images

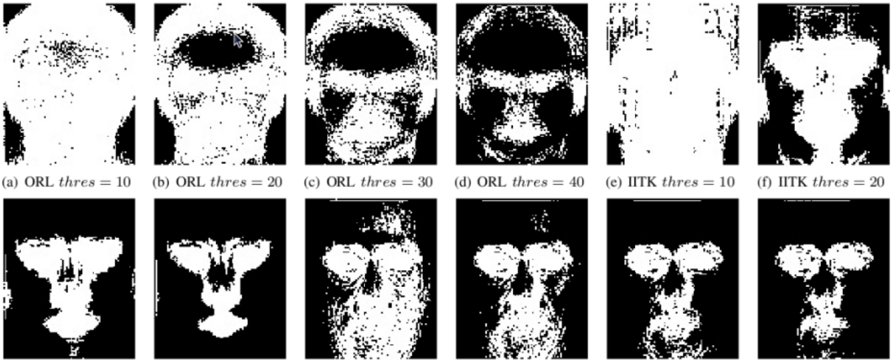
Weighing Function

Algorithm 1 *Weighing*(*database*)

Require: Binary Image W serving as weighing function given database.
Ensure: Important facial feature point have value 1 in Binary image W

```
Initialize a 2d-array A[r][c] to 0;
for all I ∈ database do
    for all i, j do
        if I[i][j] is a strong edge point then
            A[i][j] ← A[i][j] + 1;
        end if
    end for
end for
for all i, j do
    A[i][j] ← A[i][j]*255 / N;
end for
for all i, j do
    if A[i][j] > threshold then
        A[i][j] ← 1;
    else
        A[i][j] ← 0;
    end if
end for
Write this matrix A to a Binary image W;
```

Weighing Function



(a) ORL $thres = 10$ (b) ORL $thres = 20$ (c) ORL $thres = 30$ (d) ORL $thres = 40$ (e) IITK $thres = 10$ (f) IITK $thres = 20$

(g) IITK $thres = 30$ (h) IITK $thres = 40$ (i) Bern $thres = 10$ (j) Bern $thres = 20$ (k) Bern $thres = 30$ (l) Bern $thres = 40$

Figure: Weighing Function for different databases using *threshold* values such as 10,20,30,40

Notations

Parameter Description

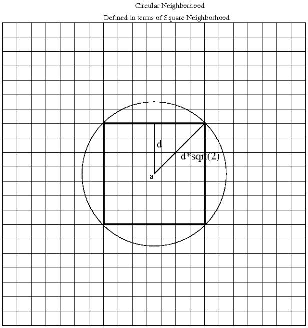
Parameter	Description
$\boldsymbol{A} \mid \boldsymbol{B}$	The corresponding gt-transformed images $(r-2)\times(c-2)$, boundary pixels are ignored;
$\boldsymbol{N_B^a}$	Neighborhood of pixel a in image B;
$\boldsymbol{V(a)}$	The 8-element vector at pixel a;
tval_a	The decimal equivalent of $V(a)$, *i.e.* the transformed value of pixel a;
$\boldsymbol{EWDM(A,B)}$	Edge Weighted Distance measure between A and B;
$\boldsymbol{p}$	Order of the norm ;
$\boldsymbol{N}$	Total number of important pixels for that database;
$\boldsymbol{N_{AB}^U}$	Total number of unmatched important pixels of A, when A is compared with B;
$\boldsymbol{Match(a,B)}$	Matches a pixel a with B, and returns 1 if Matched or 0 if Unmatched;

Defining N_B^a

- Neighborhood of pixel a in image B
- Pixel's within a distance of $d\sqrt{2}$ from pixel a is considered to be in its neighborhood.

Neighborhood

$$N_B^a = \{b \in B \mid \|a - b\| \leq d\sqrt{2}\}$$



Defining $EWDM(A, B)$

- $Match(a, B)$ is defined as:

Matching

$$Match(a, B) = \begin{cases} 1 & \text{If } \exists_{b \in N_B^a} \; V(a) = V(b) \text{ [i.e. Matched]} \\ 0 & \text{else} \end{cases}$$

- N_{AB}^U is defined as:

Important Unmatched Points

$$N_{AB}^U = \sum_{a \in A} (1 - Match(a, B))$$

- $EWDM(A, B)$ is defined as:

EWDM

$$EWDM(A, B) = \| \langle \frac{N_{AB}^U}{N}, \frac{N_{BA}^U}{N} \rangle \|_p$$

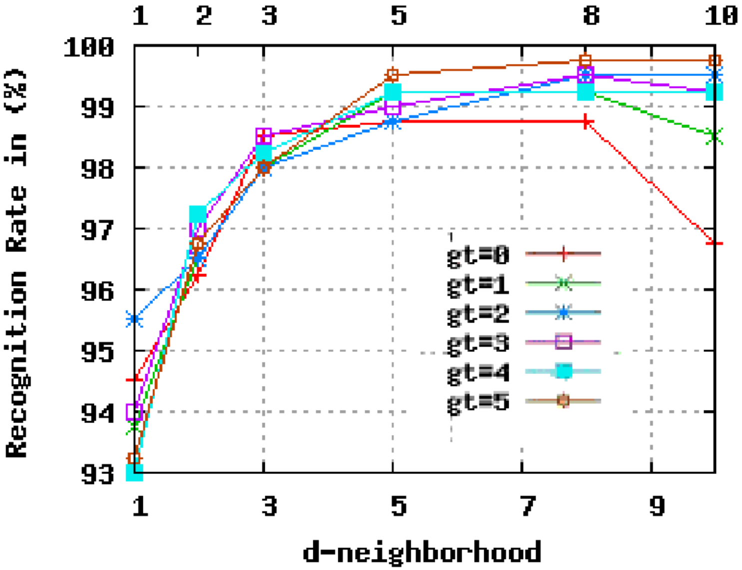
Parameterized Analysis

Parameters

- *EWDM* measure is parameterized primarily by two parameters gt and d, the third parameter p (order of norm) is set to 20 for this work.
- Gray value Tolerance gt can vary within range $[0, 5]$.
- Neighborhood parameter d can vary within range $[1, 15]$.

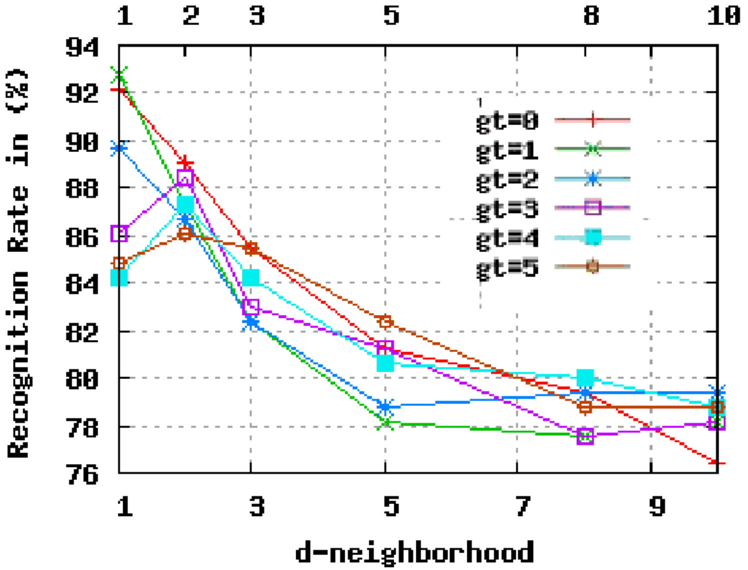
ORL:Pose and Expression Variations

ORL:top 1 [$gt = 5, d = 8, RR = 99.75\%$]



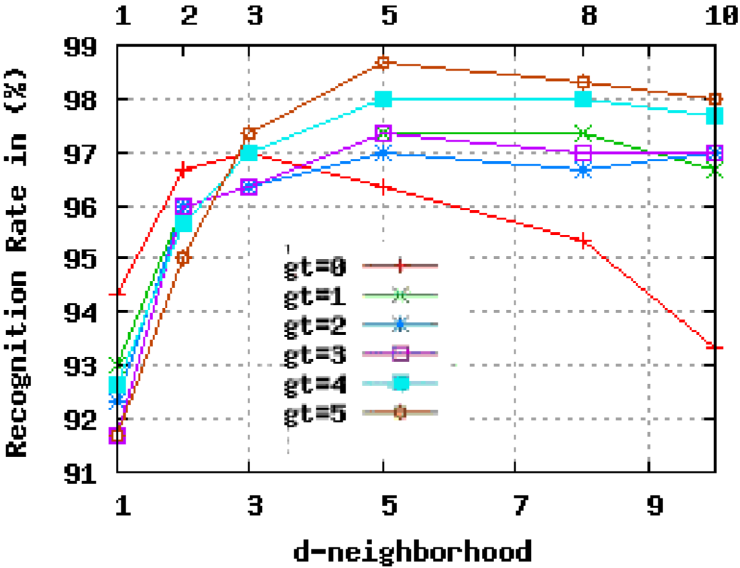
YALE:Illumination and Expression Variations

YALE:top 1 [$gt = 1, d = 1, RR = 92.75\%$]

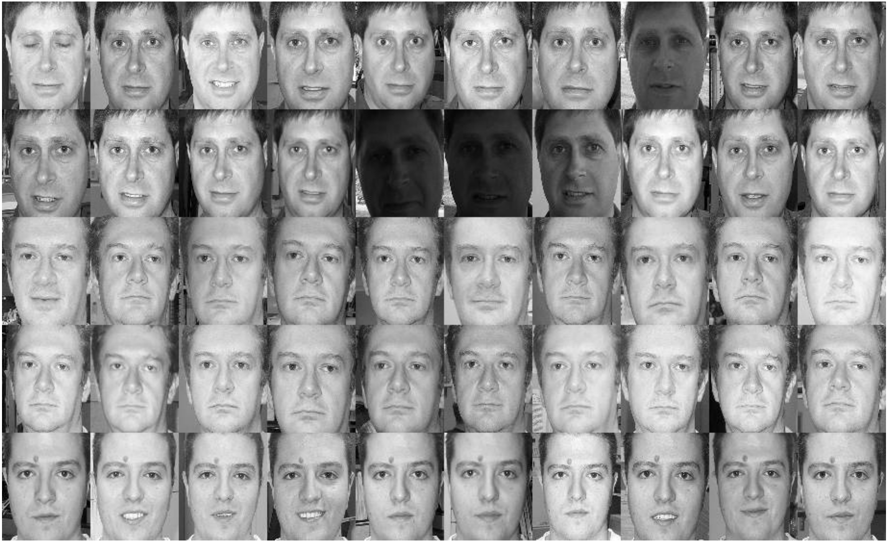


BERN:Big Pose and Expression Variations

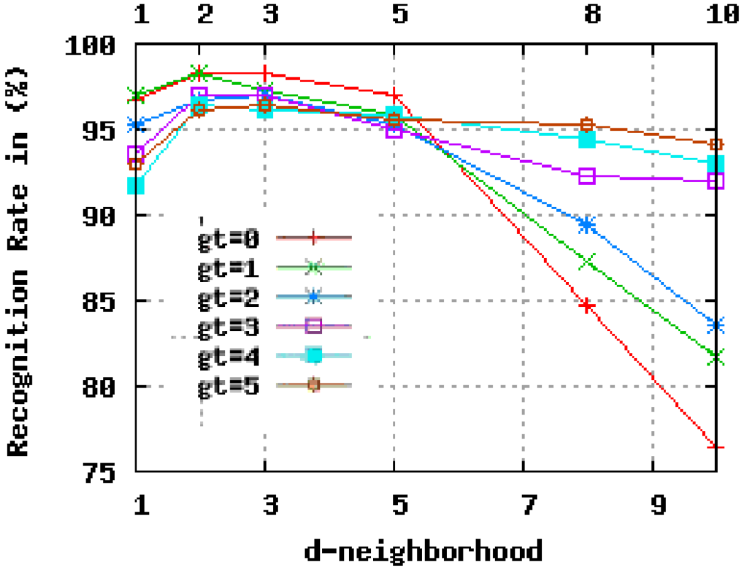
BERN:top 1 [$gt = 5, d = 5, RR = 98.66\%$]



CALTECH:Small Pose, Expression, Illumination and Background Variation



CALTECH:top 1 [$gt = 1, d = 2, RR = 98.23\%$]



IITK:Very Small Expression and Pose Variations

The best result for IITK database are with parameters $[gt = 5, d = 5]$ and recognition rate is $RR = 99.73\%$

Comparative Analysis

ORL and *YALE*

Distance Measure	Recognition rate (%) ORL	YALE
PCA	63	50
HD	46	66
PHD	72.08 ($f = 0.85$)	84 ($f = 0.7$)
M2HD	75	80
SWHD	82	82
SW2HD	88	83
SEWHD	88	85
SEW2HD	91	89
H_{pg}	91.25	83.3 ($f = 0.55$)
EWDM	**99.75** ($gt = 5,\ d = 11$)	**92.73** ($gt = 0, d = 1$)

Table: Comparative study on ORL and YALE when considering *top* 1 best match

Comparative Analysis

BERN

Test Faces	Recognition rate (%) PHD ($f = 0.85$)	LEM	H_{pg}	**EWDM** ($gt = 5, d = 5$)
Looks right/left	74.17	74.17	95.83	**99.00**
Looks up	43.33	70.00	90.00	**99.00**
Looks down	61.66	70.00	68.33	**98.00**
Average	58.75	72.09	87.50	**98.66**

Table: Comparative study on BERN database when considering *top* 1 best match

Summarized Performance of *EWDM*

Databases Vs Parameters

Db,Nor	**S,P,T**	**Time**	**gt,d,RR% [top1]**	**gt,d,RR% [top5]**	**Varying**
ORL,N	40, 10, 400	1.8	5, 8, 99.75	5, 8, 100	Poses and Expressions
YALE,Y	15, 11, 165	1.2	1, 1, 92.75	1, 2, 98.84	Illumination and Expressions
BERN,N	30, 10, 300	1.6	5, 5, 98.66	5, 5, 99.4	Poses and Expressions
CALTECH,Y	17, 20, 340	1.6	1, 2, 98.23	1, 3, 99.75	Poses and Illumination
IITK,N	149, 10, 1490	4.6	5, 5, 99.73	4, 5, 100	Poses and Scale

Table: Databases vs Parameters

Conclusion

- Edge Weighted Dissimilarity Measure (*EWDM*) measure proposed is different from existing Hausdorff distance based methods as it works on *gt*-transformed images.
- It is computationally inexpensive and provides good performance.
- Parameters gt, d, p are set taking into account the illumination variation and the nature of the images.

Discriminative Power

It has shown tolerance to varying poses, expressions and illumination conditions and can achieve a higher recognition rate than HD, PHD, MHD, M2HD, SWHD, SW2HD, SEWHD, SEW2HD, H_g and H_{pg}.

A.Samal and P.A.Iyengar,
Automatic recognition and analysis of human faces and facial expressions; a survey,
Pattern recognition 25 (1) (1992) 65-77.

R.Chellappa, C.L.Wilson and S.Sircohey,
Human and machine recognition of faces: a survey,
Proc. IEEE 83 (5) (1995) 705-740.

M.Turk and A.Pentland,
Eigenfaces for recognition,
Journal of cognitive Neuroscience, March 1991.

L.Wiskott, J.-M.Fellous, N.Kuiger and C.Von der Malsburg,
Face recognition by elastic bunch graph matching,
IEEE Tran. on Pattern Anal.Mach.Intell., 19: 775-779.

S.Lawrence, C.L.Giles, A.C.Tsoi, and A. D. Back,
Face recognition: A convolutional neural network approach,
IEEE Trans. Neural Networks, 8:98-113, 1997.

Guodong Guo, Stan Z. Li, and Kapluk Chan,
Face Recognition by Support Vector Machines,
Automatic Face and Gesture Recognition, 2000.Proc. Fourth IEEE Inter.Conf. on Volume,Issue,2000 Page(s):196-201

F.S.Samaria,
Face recognition using Hidden Markov Models. PhD thesis,
Trinity College, University of Cambridge,Cambridge,1994.

D.P.Huttenlocher, G.A.Klanderman and W.A.Rucklidge,
Comparing images using the Hausdorff distance,
IEEE Trans.Pattern Anal.Mach.Intell,vol.15, no.9,pp.850-863, sep.1993.

W.J.Rucklidge,
Locating objects using the Hausdorff distance,
ICCV 95: Proc. 5th Int. Conf. Computer Vision, Washington, D.C, June 1995, pp. 457-464.

B.Takacs,
Comparing face images using the modified Hausdorff distance,

Pattern Recognit,vol.31, no.12,pp.1873-1881, 1998.

B.Guo, K.-M.Lam, K.-H.Lin and W.-C.Siu,
Human face recognition based on spatially weighted Hausdorff distance,
Pattern Recognit. Lett., vol. 24,pp.499-507, Jan. 2003.

K.-H.Lin, K.-M.Lam and W.-C.Siu,
Spatially eigen-weighted Hausdorff distances for human face recognition,
Pattern Recognit.,vol.36,pp.1827-1834, Aug. 2003.

E.P.Vivek and N.Sudha,
Gray Hausdorff distance measure for comparing face images,
IEEE Trans. Inf. Forensics and Security, vol.1, no. 3, Sep. 2006.

N.Sudha and Y.Wong,
Hausdorff distance for iris recognition,
Proc. of 22nd IEEE Int. Symp. on Intelligent Control ISIC 2007,pages 614-619,Singapore, October 2007.

M.Dubuisson and A.K.Jain,

A modified Hausdorff distance for object Matching,
Proc. 12th Int. conf. on Pattern Recognition (ICPR), Jerusalem, Israel, (1994).

- Y.Gao and M.K.Leung,
Face recognition using line edgemap,
IEEE Trans. Pattern Anal. Machine Intell.,vol.24, pp.764-779, Jun. 2002.

- The ORL Database of Faces[Online], Available:http://www.uk.research.att.com/facedatabase.html.

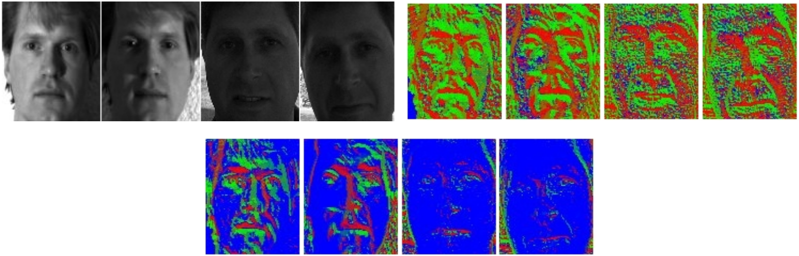
- The Yale University Face Database[Online], Available:http://cvc.yale.edu/projects/yalefaces/yalefaces.html.

- The Bern University Face Database[Online], Available:ftp://ftp.iam.unibe.ch/pub/images/faceimages/.

- The Caltech University Face Database[Online], Available:http://www.vision.caltech.edu/html-files/archive.html.

- David A. Forsyth and Jean Ponce, *Computer Vision - A Modern Approach, Pearson Education, 2003.*
- Yang,M.H.;Kriegman,D.J. and Ahuja, N,
  Detecting Faces in Images: A Survey,
  IEEE Transaction (PAMI), Vol.24, No. 1, (2002),(34-58).
- Li,S.Z and Jain, A.K
  Handbook of Face Recognition,
  Springer-Verlag, (2005)
- Yuankui Hu and Zengfu Wang,
  A Similarity Measure Based on Hausdorff Distance for Human Face Recognition,
  18^{th} International Conference on Pattern Recognition (ICPR06), IEEE (2006).
- Gary Bradski, Adrian Kaehler
  Learning OpenCV: Computer Vision with the OpenCV Library,
  [ONLINE], Available at `http://www.amazon.com/Learning-OpenCV-Computer-Vision-Library/dp/0596516134`

Big Illumination Variation [use $gt = 0$]



Effect of High *gt* values under heavy illumination variation

- With higher *gt* values more and more elements of $V(a)$ start acquiring value 1.
- This will boost the blue value of pixels in the *gt*-transformed images.
- Directional lights and heavy illumination condition variations may further lift up the blue value upto an extent that blue color starts dominating in *gt*-transformed image.

Some Properties of *EWDM* and *ewdm*

Properties

1. $EWDM(A, B) = EWDM(B, A)$.
2. If $ewdm(A, B) = K$, then $K \cdot N_a$ pixels of A do not have any pixel with same transformed value within its neighborhood in B.
3. $EWDM(A, B)$ and $ewdm(A, B)$ are always positive and normalized between 0 and 1.
4. $EWDM(A, B)$ and $ewdm(A, B)$ are parameterized by gt, d and p.

Efficient *Match(a,B)*

- Computing $EWDM(A, B)$ using naive method requires $O(r^2c^2)$ time , which is prohibitively computationally intensive.
- Performing $Match(a, B)$ operation efficiently an array of pointers to linked list BLIST is created.

BLIST

It has 3^8 elements such that $\forall i \in [0, 3^8 - 1]$ *the i^{th} element points to a linked list of pixels having the transformed value i* [14].

Date-Structure BLIST

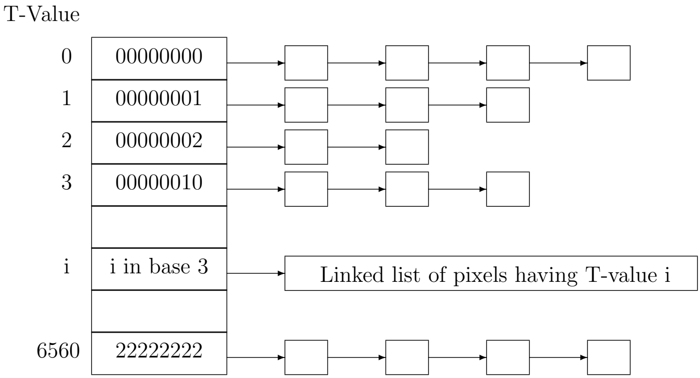


Figure: Data Structure: BLIST

Time Complexity

Preprocessing

- Gray scale images sized $r \times c$ transformed into *gt*-Transformed images. It is done once and single scan of the whole image is sufficient.
- Time complexity is $O(rc)$.

Processing

- Constructing data structure BLIST require $O(rc)$ time.
- *Match* function involves linear search of a linked list of pixels.
- Time taken by *Match* depends on the length of the list. Assuming that k is the length of the largest linked list.
- Computing *EWDM(A,B)*, $Match(a, B)$ function has to be called $2rc$ times, therefore time required to compute *EWDM* will be $O(krc)$.