A Novel Face Recognition Measure using Normalized Unmatched Points

Aditya Nigam
Supervisor: Dr. Phalguni Gupta

Department of Computer Science and Engineering
Indian Institute of Technology Kanpur

May 13, 2009
Table of contents

1 Problem Definition
   • Statement and Motivation

2 Related Work
   • HD and its Variants

3 Proposed NUP Measure
   • Pre-Processing
   • Defining NUP Measure

4 Efficient Computation of NUP
   • Algorithm
   • Analysis

5 Experimental Results and Analysis
   • Setup
   • Results

6 Conclusion
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.
- NUP 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, \(\|\cdot\|\) is the norm of the vector.
**HD Example**

![Diagram]

<table>
<thead>
<tr>
<th>Pairs of Points</th>
<th>Distances</th>
<th>Min Value and Correspondance</th>
<th>Max Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-a</td>
<td>10</td>
<td>10 (1-a)</td>
<td></td>
</tr>
<tr>
<td>1-b</td>
<td>14</td>
<td>1 corresponds to a</td>
<td></td>
</tr>
<tr>
<td>2-a</td>
<td>8</td>
<td>8 (2-a)</td>
<td>12 (3-a)</td>
</tr>
<tr>
<td>2-b</td>
<td>10</td>
<td>2 corresponds to a</td>
<td></td>
</tr>
<tr>
<td>3-a</td>
<td>12</td>
<td>12 (3-a)</td>
<td></td>
</tr>
<tr>
<td>3-b</td>
<td>15</td>
<td>3 corresponds to a</td>
<td></td>
</tr>
</tbody>
</table>

This is the worst correspondence [Most Dissimilar Points]

**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:

\[
\text{PHD}(A, B) = \text{PHD}(B, A) = \max(\text{phd}(A, B), \text{phd}(B, A))
\]

Here \(\text{phd}(A, B)\) is the directed *PHD*, which is defined by:

**Directed phd**

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

- 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 \( N_B^a \): 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

\[
\text{swhd}(A, B) = \max_{a \in A} \left[ w(b) \cdot \min_{b \in B} \|a - b\| \right]
\]

\[
\text{sw2hd}(A, B) = \frac{1}{N_a} \sum_{a \in N_a} \left[ w(b) \cdot \min_{b \in B} \|a - b\| \right]
\]
```
**Spatial Weighing Function**

Where \( w(x) \) is defined as:

\[
w(x) = \begin{cases} 
1 & x \in \text{Important facial region} \\
W & x \in \text{Unimportant facial region} \\
0 & x \in \text{Background region}
\end{cases}
\]
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.
Figure: Eigenfaces
Defining \textit{SEWHD} and \textit{SEW2HD}

- Proposed \textit{SEWHD} and \textit{SEW2HD} [12] are defined as:

**Directed sewhd and sew2hd**

\[
\text{sewhd}(A, B) = \max_{a \in A} \left[ w_e(b) \cdot \min_{b \in B} \|a - b\| \right]
\]

\[
\text{sew2hd}(A, B) = \frac{1}{N_a} \sum_{a \in N_a} \left[ w_e(b) \cdot \min_{b \in B} \|a - b\| \right]
\]

where \(w_e(x)\) is defined as:

\(w_e(x) = \text{Eigen weight function generated by the first eigen vector}\)
$H_g$ and $H_{pg}$

- Edge images loose most of the important facial features, which are very useful for facial discrimination.
- $H_g$ and $H_{pg}$ [13] measures works on quantized images and are found robust to slight variation in poses, expressions and illumination.

Quantized Images

Images with $n \geq 5$ retains the perceptual appearance and the intrinsic facial feature information that resides in gray values (as shown in Figure below).

Figure: Quantized-faces
Defining $H_g$ and $H_{pg}$

- $H_g$ and $H_{pg}$ are defined on quantized gray images as:

**Directed $h_g$ and $h_{pg}$**

$$h_g(A, B) = \max_{a \in A_i} \max_{i=0..2^n-1} d(a, B_i)$$

$$h_{pg}(A, B) = K^{th} \max_{a \in A_i} \max_{i=0..2^n-1} d(a, B_i)$$

where $d(a, B_i)$ is defined as:

$$d(a, B_i) = \begin{cases} 
  \min_{b \in B_i} \|a - b\| & \text{if } B_i \text{ is non-empty} \\
  L & \text{otherwise}
\end{cases}$$

$A_i$ and $B_i$ are the set of pixels in quantized images $A$ and $B$ having quantized gray value $i$. 

**NUP Measure**

- *NUP* measure can be applied on \(gt\)-transformed images obtained from gray-scale facial images.
- *NUP* measure is similar to the *HD* based measures but is computationally less expensive and more accurate.
- *NUP* 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.
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.
Gray levels are hardly distinguishable (Similar) within a range of ±5 units.
Basic Comparator

\[
\begin{align*}
&= X \\
&X < \alpha \in (X, 255] \\
&> \alpha \in [0, X)
\end{align*}
\]

Where \( \alpha \) is gray value tolerance, \( \alpha \geq 0 \).

\( X \) is a gray level not merely a number.

Gray level \( X \) is neither greater than gray level \( (X - 1) \) nor less than \( (X + 1) \); ideally they should be considered as similar.
Improvement

- **Basic Comparator**
  \[
  X \begin{cases}
  = X & \text{if } \alpha \in [0, X) \\
  < \alpha \in (X, 255] & \text{if } \alpha \in (X, 255] \\
  > \alpha \in [0, X) & \text{if } \alpha \in [0, X) 
  \end{cases}
  \]

- **gt-Comparator**
  \[
  X \begin{cases}
  = \alpha \in [(X - gt), (X + gt)] & \text{if } \alpha \in [(X - gt), (X + gt)] \\
  < \alpha \in (X + gt, 255] & \text{if } \alpha \in (X + gt, 255] \\
  > \alpha \in [0, X - gt) & \text{if } \alpha \in [0, X - gt) 
  \end{cases}
  \]

Where \( gt \) is gray value tolerance, \( gt \geq 0 \).
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

REGION : LESS THAN X

X

REGION : GREATER THAN X

Figure: Basic Comparator

REGION : LESS THAN X

X

REGION : EQUAL TO X

 REGION : GREATER THAN X

Figure: \textit{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 $< RED \text{ i.e.}[0]$, Equal To $= BLUE \text{ i.e}[1]$, Greater Than $> GREEN \text{ i.e.} [2]$.

Figure: gt-Transformed images
## Notations

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A \mid B$</td>
<td>The corresponding $gt$-transformed images $(r-2) \times (c-2)$, boundary pixels are ignored;</td>
</tr>
<tr>
<td>$N^a_B$</td>
<td>Neighborhood of pixel $a$ in image $B$;</td>
</tr>
<tr>
<td>$V(a)$</td>
<td>The 8-element vector at pixel $a$;</td>
</tr>
<tr>
<td>$tval_a$</td>
<td>The decimal equivalent of $V(a)$, i.e. the transformed value of pixel $a$;</td>
</tr>
<tr>
<td>$NUP(A, B)$</td>
<td>Undirected Normalized Unmatched Points measure between $A$ and $B$;</td>
</tr>
<tr>
<td>$nup(A, B)$</td>
<td>Directed Normalized Unmatched Points measure, when $A$ is compared with $B$;</td>
</tr>
<tr>
<td>$p$</td>
<td>Order of the norm;</td>
</tr>
<tr>
<td>$N_a$</td>
<td>Total number of pixels in image $A$;</td>
</tr>
<tr>
<td>$N^U_{AB}$</td>
<td>Total number of unmatched pixels of $A$, when $A$ is compared with $B$;</td>
</tr>
<tr>
<td>$Compare(A, B)$</td>
<td>Compares image $A$ to image $B$, and returns $N^U_{AB}$;</td>
</tr>
<tr>
<td>$Match(a, B)$</td>
<td>Matches a pixel $a$ with $B$, and returns 1 if Matched or 0 if Unmatched;</td>
</tr>
</tbody>
</table>
Defining $N^a_B$

- 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**

\[ N^a_B = \{ b \in B \mid \|a - b\| \leq d \sqrt{2} \} \]
Defining $\text{Compare}(A, B)$ and $N_{AB}^U$

- $\text{Compare}(A, B)$ compares two $gt$-transformed images $A$ and $B$.
- **Returns** $N_{AB}^U$ (*i.e.* Total number of unmatched pixels of $A$, when $A$ is compared with $B$), defined as:

\[
N_{AB}^U = \sum_{a \in A} (1 - \text{Match}(a, B))
\]
Defining \( \text{Match}(a, B) \)

- \( \text{Match}(a, B) \) matches a pixel \( a \) with a \( gt \)-transformed image \( B \).
- **Returns** 1 if there is a pixel within the neighborhood of \( a \) in image \( B \), having same \( gt \)-transformed value (i.e. Matched), Else **Returns** 0 (i.e. Unmatched).

- \( \text{Match}(a, B) \) can be defined as:

\[
\text{Match}(a, B) = \begin{cases} 
1 & \text{If } \exists b \in N_B^a \ V(a) = V(b) \text{ [i.e. Matched]} \\
0 & \text{else}
\end{cases}
\]
Defining $NUP(A, B)$ and $nup(A, B)$

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

<table>
<thead>
<tr>
<th>Undirected NUP</th>
</tr>
</thead>
<tbody>
<tr>
<td>$NUP(A, B) = | \langle nup(A, B), nup(B, A) \rangle |_p$</td>
</tr>
</tbody>
</table>

where $nup(A, B)$ is defined as:

<table>
<thead>
<tr>
<th>Directed nup</th>
</tr>
</thead>
<tbody>
<tr>
<td>$nup(A, B) = \frac{N_{AB}^U}{N_a}$</td>
</tr>
</tbody>
</table>

and $\| . \|_p$ is the $p^{th}$ norm.
Some Properties of $NUP$ and $nup$

Properties

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

Aditya Nigam (M.Tech. CSE)  
Normalized Unmatched Points  
May 13, 2009  
30 / 58
Efficient $\text{Match}(a, B)$

- Computing $NUP(A, B)$ using naive method requires $O(r^2c^2)$ time, which is prohibitively computationally intensive.
- Performing $\text{Match}(a, B)$ 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

T-Value

<table>
<thead>
<tr>
<th></th>
<th>00000000</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>00000000</td>
<td>1</td>
<td>00000001</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>00000010</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>i</td>
<td>i in base 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6560</td>
<td>22222222</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Linked list of pixels having T-value i

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 \( NUP(A,B) \), \( Match(a,B) \) function has to be called \( 2rc \) times, therefore time required to compute \( NUP \) will be \( O(krc) \).
Figure: Images produced after various phases
Testing Strategy

- Whole database is treated as the testing set, then each image of the testing set is matched with all other images excluding itself. Finally top $n$ best matches are reported.
- Match is announced if and only if a subject’s image got matched with another pose of himself/herself.

### Recognition Rate

$$\text{Recognition rate} = \frac{\text{Number of matches}}{(\text{Total number of images}) \times n}$$
Parameterized Analysis

Parameters

- NUP 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 \([1, 15]\).

Databases Vs Parameters

<table>
<thead>
<tr>
<th>Db,Nor</th>
<th>S,P,T</th>
<th>Time</th>
<th>gt,d,RR% [top1]</th>
<th>gt,d,RR% [top5]</th>
<th>Varying</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORL,N</td>
<td>40, 10, 400</td>
<td>1.8</td>
<td>5, 8, 99.75</td>
<td>5, 10, 90.15</td>
<td>Poses and Expressions</td>
</tr>
<tr>
<td>YALE,Y</td>
<td>15, 11, 165</td>
<td>1.2</td>
<td>1, 1, 92.75</td>
<td>0, 2, 85.57</td>
<td>Illumination and Expressions</td>
</tr>
<tr>
<td>BERN,N</td>
<td>30, 10, 300</td>
<td>1.6</td>
<td>5, 5, 98.66</td>
<td>5, 8, 75.80</td>
<td>Poses and Expressions</td>
</tr>
<tr>
<td>CALTECH,Y</td>
<td>17, 20, 340</td>
<td>1.6</td>
<td>1, 2, 98.23</td>
<td>0, 2, 95.64</td>
<td>Poses and Illumination</td>
</tr>
<tr>
<td>IITK,N</td>
<td>149, 10, 1490</td>
<td>4.6</td>
<td>5, 5, 99.73</td>
<td>4, 5, 99.58</td>
<td>Poses and Scale</td>
</tr>
</tbody>
</table>

Table: Databases vs Parameters
Big Illumination Variation \([\text{use } gt = 0]\)

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.
ORL: Pose and Expression Variations
ORL: top 1 \( [gt = 5, d = 8, RR = 99.75\%] \)
ORL: top 5 \( [gt = 5, \ d = 10, \ RR = 90.15\%] \)
YALE: Illumination and Expression Variations
YALE: top 1 \( [gt = 1, d = 1, RR = 92.75\%] \)
YALE: top 5 \[gt = 0, d = 2, RR = 85.57\%\]
BERN: Big Pose and Expression Variations
BERN: top 1 \[gt = 5, d = 5, RR = 98.66\%\]
BERN: top 5 \[ gt = 5, \ d = 8, \ RR = 75.80\% \]
CALTECH: Small Pose, Expression, Illumination and Background Variation
CALTECH: top 1 \[gt = 1, d = 2, RR = 98.23\%\]
CALTECH: top 5 \[gt = 0, d = 2, RR = 95.64\%\]
IITK: Very Small Expression and Pose Variations
IITK: top 1 \([gt = 5, d = 5, RR = 99.73\%]\)
IITK: top 5 \([gt = 4, d = 5, RR = 99.58\%]\)
## Comparative Analysis

### ORL and YALE

<table>
<thead>
<tr>
<th>Distance Measure</th>
<th>Recognition rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCA</td>
<td>ORL: 63</td>
</tr>
<tr>
<td>HD</td>
<td>ORL: 46</td>
</tr>
<tr>
<td>PHD</td>
<td>PHD: 72.08 ($f = 0.85$)</td>
</tr>
<tr>
<td>M2HD</td>
<td>ORL: 75</td>
</tr>
<tr>
<td>SWHD</td>
<td>ORL: 82</td>
</tr>
<tr>
<td>SW2HD</td>
<td>ORL: 88</td>
</tr>
<tr>
<td>SEWHD</td>
<td>ORL: 88</td>
</tr>
<tr>
<td>SEW2HD</td>
<td>ORL: 91</td>
</tr>
<tr>
<td>$H_{pg}$</td>
<td>ORL: 91.25</td>
</tr>
<tr>
<td><strong>NUP</strong></td>
<td><strong>99.75 ($gt = 5, d = 11$)</strong></td>
</tr>
</tbody>
</table>

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

### BERN

<table>
<thead>
<tr>
<th>Test Faces</th>
<th>Recognition rate (%)</th>
<th>PHD ($f = 0.85$)</th>
<th>LEM</th>
<th>$H_{pg}$</th>
<th>NUP ($gt = 5, d = 5$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Looks right/left</td>
<td>74.17</td>
<td>74.17</td>
<td>95.83</td>
<td>99.00</td>
<td></td>
</tr>
<tr>
<td>Looks up</td>
<td>43.33</td>
<td>70.00</td>
<td>90.00</td>
<td>99.00</td>
<td></td>
</tr>
<tr>
<td>Looks down</td>
<td>61.66</td>
<td>70.00</td>
<td>68.33</td>
<td>98.00</td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>58.75</td>
<td>72.09</td>
<td>87.50</td>
<td>98.66</td>
<td></td>
</tr>
</tbody>
</table>

Table: Comparative study on BERN database when considering top 1 best match
## Overall Analysis

### Overall

<table>
<thead>
<tr>
<th>Top-n</th>
<th>ORL</th>
<th>YALE</th>
<th>CALTECH</th>
<th>BERN</th>
<th>IITK</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>99.75</td>
<td>92.72</td>
<td>98.23</td>
<td>98.66</td>
<td>99.73</td>
</tr>
<tr>
<td>2</td>
<td>98.63</td>
<td>89.7</td>
<td>98.08</td>
<td>89.33</td>
<td>99.73</td>
</tr>
<tr>
<td>3</td>
<td>97.10</td>
<td>88.11</td>
<td>97.25</td>
<td>83.77</td>
<td>99.66</td>
</tr>
<tr>
<td>4</td>
<td>94.87</td>
<td>86.51</td>
<td>96.40</td>
<td>79.41</td>
<td>99.63</td>
</tr>
<tr>
<td>5</td>
<td>90.15</td>
<td>85.57</td>
<td>95.64</td>
<td>75.80</td>
<td>99.58</td>
</tr>
<tr>
<td>6</td>
<td>86.13</td>
<td>83.23</td>
<td>94.46</td>
<td>71.33</td>
<td>99.55</td>
</tr>
<tr>
<td>7</td>
<td>82.10</td>
<td>79.74</td>
<td>93.27</td>
<td>66.57</td>
<td>99.41</td>
</tr>
<tr>
<td>8</td>
<td>78.50</td>
<td>73.11</td>
<td>92.42</td>
<td>62.12</td>
<td>99.14</td>
</tr>
<tr>
<td>9</td>
<td>74.01</td>
<td>67.20</td>
<td>91.30</td>
<td>57.70</td>
<td>98.05</td>
</tr>
</tbody>
</table>

Table: Overall Analysis (considering top-n best matched)
Figure: NUP measure on different face databases, top n best matches.
Future Work

- In constrained environment which is uniformly well illuminated \textit{NUP} measure could also be used for video surveillance, scene segmentation in videos, face detection, face authentication.

Fast First Level Scanner

- For recognition in complex varying environments with big images it can also be used as fast first level scanner, working on under sampled images providing assistance to the higher levels.

- It can also be extended to other biometric traits as iris and ear.
Conclusion

- Normalized Unmatched Points (NUP) 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*,

R. Chellappa, C. L. Wilson and S. Sircohey,
*Human and machine recognition of faces: a survey*,

M. Turk and A. Pentland,
*Eigenfaces for recognition*,

L. Wiskott, J.-M. Fellous, N. Kuiger and C. Von der Malsburg,
*Face recognition by elastic bunch graph matching*,

S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back,
*Face recognition: A convolutional neural network approach*,
Guodong Guo, Stan Z. Li, and Kapluk Chan,
*Face Recognition by Support Vector Machines*,

F.S. Samaria,

D.P. Huttenlocher, G.A. Klanderman and W.A. Rucklidge,
*Comparing images using the Hausdorff distance*,

W.J. Rucklidge,
*Locating objects using the Hausdorff distance*,

B. Takacs,
*Comparing face images using the modified Hausdorff distance,*
B. Guo, K.-M. Lam, K.-H. Lin and W.-C. Siu,
*Human face recognition based on spatially weighted Hausdorff distance*,

K.-H. Lin, K.-M. Lam and W.-C. Siu,
*Spatially eigen-weighted Hausdorff distances for human face recognition*,

E. P. Vivek and N. Sudha,
*Gray Hausdorff distance measure for comparing face images*,

N. Sudha and Y. Wong,
*Hausdorff distance for iris recognition*,

M. Dubuisson and A. K. Jain,
A modified Hausdorff distance for object Matching, 

Y.Gao and M.K.Leung, 
*Face recognition using line edgemap*, 


The Bern University Face Database[Online],


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