AN EFFICIENT TECHNIQUE TO COMPUTE MODIFIED HAUSDORFF DISTANCE IN HUMAN FACE RECOGNITION

Bui Thanh Tinh (1), Truong Thien Nhan (2), Dang Nguyen Chau (3)
Department of Telecommunications Engineering, HCM University of Technology
[1]1513505@hcmut.edu.vn, [2]1512281@hcmut.edu.vn, [3]chaudn@hcmut.edu.vn

ABSTRACT

The Hausdorff Distance (HD) between two point sets is a distance metric in similarity measures and widely used in Computer Vision, and Artificial Intelligence. Modified Hausdorff Distance (MHD), which is a version of the HD, is used for comparing efficiently point sets extracted from two objects and robust to various noise. Nevertheless, because of its inherent computational complexity, it is difficult for computing normally the MHD with the huge database. Thus, we propose an efficient technique for computing quickly the MHD with acceptable error. Different from the normal way computed by using all of the dominant points in the point sets, our idea creates binary images having various sizes with dominant points and saves the coordinates of these dominant points using matrices. Then, we compute approximately the MHD using these matrices in ascending order of the size of the binary images corresponding to the matrices. Our technique achieves performance better than the normal one while the complexity of computing is lower than the initial technique by 50-70 orders of magnitude in face recognition. Experiments demonstrate the efficiency of the proposed technique.

Keywords: *Modified Hausdorff Distance, point sets, computational complexity, face recognition.*

1. INTRODUCTION

Face recognition (FR), the process of identifying people through face images, has been an active research area for the last 20 years. FR system identifies a face by matching the image of this face with the most similar image in the database. FR has many practical applications, such as security, access control, person identification, and surveillance systems (M. Sharif, et al., 2017). However, in most systems, matching is the most computationally expensive operation because of the large number of available images in the database. The efficient search method is the prerequisite of identification systems, especially FR systems.

The Hausdorff Distance (HD) is an important metric that is widely used in many domains, including Image Processing, Pattern Recognition, Shape Matching, and Artificial Intelligence. In face recognition, most images contain various levels of noise and the HD is sensitive to these. The presence of a few outlying points can cause a high error in the computation of the HD whereas the two objects may be very similar. A modified Hausdorff distance (MHD), which obtained more desirable performance for object matching, was proposed to alleviate the sensitivity problem (Dubuisson & Jain, 1994). However, the existing algorithm for computing the MHD have high complexity, it is restricted in face matching.

The main contributions and advantages of this paper include: A proposed technique combines resizing the sizes of the images, classifying the locals and marking the potential ones to decrease the computational complexity of computing the MHD. A binary matrix containing the dominant points is resized multiple times using division to create matrices having various sizes. Then, they are used for computing the MHD depended on the proposed technique.

The rest of this paper is organized as follows. Section 2 presents previous works related to the applications of the MHD in face recognition and algorithms improved the HD. Section 3 describes mathematical formulas and theories to formulate the problem. The proposed technique to compute efficiently the MHD is presented in details in Section 4. The experimental results and comparisons with the original technique are shown in Section 5. Finally, Section 6 derives the paper’s conclusions.

2. RELATED WORK

The Hausdorff Distance (HD) is a useful measurement to determine the similarity between two point sets.

Thus, it is commonly used in object matching. The first method (Huttenlocher et al., 1993) used the HD for comparing images. The Kth ranked value in the set of distances is chosen to be the value of the HD instead of 1st ranked value. Takács is the first to lay the groundwork for face recognition using the MHD (Takács, 1998). After using edge filter methods, he used all pixels in the edges to compute the MHD. However, many pixels in the edges have similar properties and don't affect the recognition. A new method (Gao, 2003) was proposed to compute the MHD using dominant points in the edges. This method is 28 times faster than that of the Takács's method. Face recognition and detection is done by using the HD with SURF and SVM (Pawana & Sachin, 2016).

Recently, some efficient algorithms for HD computation have been proposed with the purpose of reducing the computational complexity of the HD. The novel technique (Nutanong, et al. (2011)) utilizes hierarchical indexes based on an R-tree. The EB algorithm (Taha & Hanbury 2015) achieves high efficiency in processing medical images. The LSS algorithm (Chen, et al. 2017) uses points processed by the Morton Curve for computing the HD and is 3 times faster than the EB. However, we observe that there still isn't an algorithm to optimize computing the MHD.

3. PROBLEM FORMULATION

3.1 The modified Hausdorff distance

Named after Felix Hausdorff, the Hausdorff distance is a metric between two point sets. Unlike other common metrics, the HD measures the degree of resemblance of two point sets (a many-to-many correspondence) instead of forming a one-to-one mapping between the two. Given two finite point sets $A = \{a_1, a_2, \dots, a_m\}$ and $B = \{b_1, b_2, \dots, b_n\}$, the HD is defined as

$$H(A,B) = \max(\mathrm{h(A,B)}, \mathrm{h(B,A)}) \tag{1}$$

where

$$h(A,B) = \max_{a\in A} \min_{b\in B} \|a-b\| \tag{2}$$

and $\|.\|$ is the Euclidean norm on the points of A and B. The function $h(A, B)$ is called the directed Hausdorff distance from A to B.

The HD is very sensitive to even a single outlying point of A or B. Dubuisson & Jain (1994) proposed 24 different forms of Hausdorff distance and proved that the MHD has the best performance. The formal of the directed MHD between A and B is as follows

$$h_{MHD}(A,\mathrm{B}) = \frac{1}{M}\sum_{a_i\in A} \min_{b_j\in B} \|\mathrm{a}_i - \mathrm{b}_j\| \tag{3}$$

where M is the number of points in A. The MHD is defined similarly to (1). Conspicuously, the MHD has the characteristic of average from formula (3). Typically, because of the insignificant number of noise points, the MHD can alleviate the sensitivity of HD to these.

3.2 Dominant point detection

The edges in an image reflect large local intensity changes that are the consequence of the geometrical structure of the object, the characteristics of the surface and the viewing point. Experimental results from physiology and psychophysics have shown that the edges of an object contain important information about its individual shape and structure and they suit to be used in face recognition (Brunelli & Poggio, 1993). The edge that is easily extracted by basic filtering algorithms is highly effective in identifying. There are many ways to find edges in a photograph, each with different advantages and disadvantages, but no method is completely optimal. In the study, we use the LEMExpression.exe program written by Y.Gao to create face edge maps. In this program, the edge detector bases on the algorithm of Nevatia and Babu (1980), then thin edges to create one-pixel-wide edges. The Dynamic-two-Strip algorithm (Dyn2S) (Leung & Yang, 1990) is utilized to detect dominants points. They are points of high curvature and more influential in the recognition than others. In experiments of Section 4, we use these points for computing the MHD. An illustrative example of the face edge map is shown in Fig. 1.

Fig. 1. A face edge map of a face image in Bern database.

3.3 The initial technique

Prior to the introduction of our technique, we first describe the original technique for the MHD calculation. It is shown in *Algorithm 1*. Apparently, there are two loops (outer loop and inner loop) and all calculations must be scan through all points. Thus, the computational complexity of *Algorithm 1* is $O(nA*nB)$, where nA and nB are the point counts of A and B.

Algorithm 1 Computing the directed MHD.

Require: Two finite point sets A and B, nA and nB are the point counts of A and B.
Ensure: The directed MHD $h(A,B)$.

```
sumA ← 0
for all ai ∈ A do
    n ←0
    for all bj ∈ B do
        n ←n+1
        distA(n) ← dist(ai , bj)
    end for
    sumA ← sumA + min(distA)
end for
hAB ← sumA/nA
return hAB
```

4. THE PROPOSED TECHNIQUE

The MHD is an AVERAGE-MIN distance that contains both average and minimization. To compute the MHD, the number of iterations of the outer loop in *Algorithm 1* can't be reduced. Therefore, the only solution to reduce the computational complexity is to reduce the average number of iterations in the inner loop. Our technique gets the idea from resizing image. The distance between two points in the image *I* is denoted by *d*. If the size of the image *I* is decreased by $n*n$ ratio with maintained aspect ratio, the distance between two respective points is d/n.

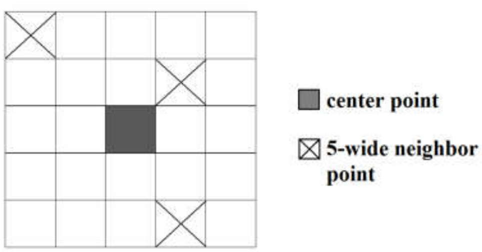

Fig. 2. Local 5*5 with center point and 5-wide neighbor points.

4.1 Data Preprocessing

Before the introduction of our technique, we first define the concepts. Consistently, the following definitions are used throughout this paper:

- *Center point*: the center of square local $w*w$.
- *Dominant points (dps)*: the points which correspond to black pixels in binary image.
- *w-wide neighbor point*: the dominant point in square local $w*w$ is around center point. Fig. 2 shows an example in detail.
- A_{m*m}: the matrix with size $m*m$ corresponds to the binary image with size $m*m$.

The face images with size $m*m$ in the database are passed through LEMExpression.exe to create binary images with dominant points. Then, the matrices corresponding to these binary images are resized multiple times using $k*k$ division. In this paper, we implement with $k = 2$ to create binary matrices with the size: (m/2)*(m/2), (m/4)*(m/4), (m/8)*(m/8), etc. The main steps to resize a matrix with size $m*m$ using $k = 2$ are presented as follows:

Step 1: Create a matrix A_{m*m}: the value of the elements at the coordinates corresponding to dominant points in the binary image with size $m*m$ equals to 1; the rest of the elements have the value to equal to 0.

Step 2: Create a matrix $A_{(m/2)*(m/2)}$. Consider point a_i with coordinate (r_i, c_i) in $A_{(m/2)x(m/2)}$, let $rr_i = r_i * 2$ and $cc_i = c_i * 2$; In local $w*w$ (w: odd) with center point (rr_i, cc_i) in A_{m*m}, if there is a or more a dominant point, the value of point a_i equals to 1 and vice versa equals to 0.

Step 3: Repeat step 2 with size *(m/4)x(m/4), (m/8)x(m/8)*, etc. In this paper, the process is over when the size of the matrix is lower than 4*4.

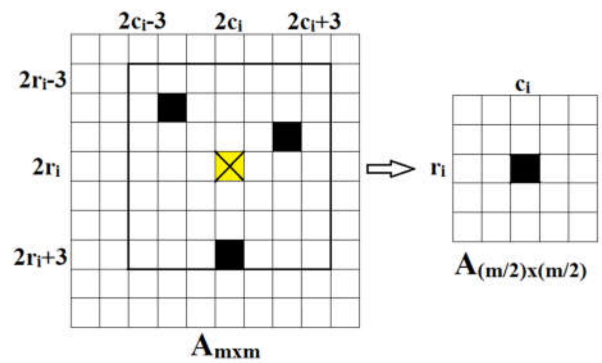

Fig. 3. How to resize a matrix with local 7*7.

4.2 Potential local search

In this subsection, we present our technique to compute the MHD using data after preprocessing. To facilitate comprehension, we compute specifically the MHD with the binary image with initial size *160*160* and the matrices have size: *80*80, 40*40, 20*20, 10*10, 5*5*. Assume that we need to compute the minimum Euclidean distance from point $a_i(r_a, c_a)$ in *A* to *B*, the process is implemented ascending order of the size of these matrices. Point (r_i, c_i) in matrix $B_{(m/2)*(m/2)}$ has the coordinate corresponding to point a_i (therefore $r_i = \lceil r_a/(160/(m/2)) \rceil$ and $c_i = \lceil c_a/(160/(m/2)) \rceil$) and the value to equal to 1, illustrated in Fig. 3.

- If the value of point $(2r_i, 2c_i)$ in matrix B_{m*m} equals to 0, we compute the minimum Euclidean distance *cmin* from w-wide neighbor points of this point to the point with the coordinate $r = \lceil r_a/(160/m) \rceil$; $c = \lceil c_a/(160/m) \rceil$ and $d = c\min*160\,/\,m$.
- If this value equals to 1, we repeat with next larger matrix until the value 0 and implement the previous case. If the size equals to *80*80* with the value 1 of the center point, we compute the minimum Euclidean distance *cmin* from point $a_i(r_a, c_a)$ to w-wide neighbor points correspond-

ing to the dominant points in matrix $B_{160*160}$. *Algorithm 2* shows this process in detail.

Implement similarly with all of the points in matrix A and average these distances to create the directed MHD $h_{MHD}(A, B)$.

Algorithm 2 Computing the minimum Euclidean distance from point $a_i(ra, ca)$ in A to B.

Require: Two finite point sets A and B, nA and nB are the point counts of A and B. Point a_i in A has the coordinate (ra, ca), $i = \overline{1:\text{nA}}$. The matrices are saved orderly: (6) 1 matrix 5x5, (5) 1 matrix 10x10, (4) 1 matrix 20x20, (3) 1 matrix 40x40, (2) 1 matrix 80x80, (1) 1 matrix 160x160.

Ensure: The $h(a_i, B)$.

```
(ra,ca) ∈ A160*160, B1 ← B160*160, B2 ← B80*80, B3
   ← B40*40, B4 ← B20*20, B5 ← B10*10, B6 ← B5*5,
   dist_ai ← 0, w ←5
for nmb = 5; nmb > 0; nmb-- do
   ri ← ceil(ra/2^nmb), ci ← ceil(ca/2^nmb)
   if Bnmb+1(ri,ci)=1 then
      r ←2*ri, c ←2*ci, control ← nmb-1
   else
      break
   end if
end for
if control = 0 then
   C ← (dps∈ B1(r-w:1:r+w, c-w:1:c+w)), n ← 0
   for all c ∈ C do
      n ← n+1
      dist_ai(n) ← dist(c, (ra, ca))
   end for
else if nmb = 5 then
   dist_ai(1) ← 1
else
   C ← (dps ∈ Bnmb+1(r-w:1:r+w, c-w:1:c+w)), n ← 0
   for all c ∈ C do
      n ← n+1
      dist_ai(n) ← dist(c, (ri, ci))
   end for
end if
cmin ← min(dist_ai)
if control ≠ 0 then
   cmin ← cmin*2^nmb
end if
return cmin
```

Because of reducing the computational complexity by estimating quickly the MHD, we must accept the error between the current result and the result computed by the original algorithm. Gao & Leung (2002) added some components to the HD to minimize this error. Apply this with the MHD, we find the sum nhcA of the number of the points in matrix A which have the distance to B to be lower than K. A high confident point ratio of A (RA) is defined as

$$R_A = \frac{n_{hcA}}{n_A} \quad (4)$$

A complete version of the MHD integrated with number disparity in formula (4) is

$$H^*_{MHD}(A,B) = \sqrt{H^2_{MHD}(\text{A},\text{B}) + (\text{W}_n\text{D}_n)^2} \quad (5)$$

where $H_{MHD}(A, B)$ is the undirected MHD defined similarly in (1) and W_n is the weight of number disparity D_n. The number disparity is defined as the average ratio of the number of the points which have the distance to another matrix to be higher than K.

$$D_n = 1 - \frac{R_A + R_B}{2} \quad (6)$$

5. EXPERIMENTS AND RESULTS

In this section, a system performance survey, which covers the various conditions of human face recognition, has been conducted. They are face recognition taken under ideal condition, varying lighting condition and varying pose. The system performances are compared with the initial technique presented in subsection 3.3.

In this study, two face databases were tested. In the database from the University of Bern (BERN, 2002), there are 30 different people. Each person has 10 gray-level images (two portraits, two looking to the right, two looking to the left, two looking downwards, and two looking upwards). In the AR database (Martinez & Benavente, 1998), there are 100 different people (50 men and 50 women). However, some images were found lost or corrupted after downloading through the Internet, the performance in the next subsections can be wrong from 1 to 2%. Each person was taken while wearing a scarf, glasses or showing different emotions. All of the images have size 160*160.

In the simulations, we choose the parameters: local 5*5 for the 5-wide neighbor points (w=5), W_n = 30, K_n = 4. The parameters w, W_n and K_n with the best performance in our experiments after we change w = {3, 5, 7}, W_n = $\overline{10:50}$ with step 10 and K_n = $\overline{3:6}$ with step 1. They can seem like the specific parameters with the used database.

5.1 Face recognition under ideal condition

In this subsection, the portraits in Bern database and AR database were used for computing the performance of the initial MHD and the proposed MHD. In these databases, each person has two gray-level face images taken in the ideal condition, one was used as the model image and another was the test image. Therefore, we used 30 pairs of frontal face images from Bern database and 100 pairs from AR database. The recognition results for the top1 match are summarized in Table 1.

Table 1. Performance under ideal condition.

Database	Technique	
	MHD	Proposed MHD
Bern	100%	100%
AR	70%	82%

Both the MHD and the proposed MHD achieved 100 percent accuracy for verifying faces in the database of Bern University. However, the proposed MHD performed better than the initial MHD by 12 percent (i.e., it correctly verified 82 percent of 100 input faces). This difference was created by the confident parameters (Wn, Dn) in Formula 5, and each pair of the values of these parameters only suits the specified database. On the other hand, the accuracy in Bern database outperformed that one in AR database, because a pair of face images from Bern database was taken with a short interval, whereas the two faces in AR database were taken with a two-week interval.

5.2 Face recognition under varying lighting conditions

In the experiments, we used 100 frontal face images of 100 people in AR database for training. Each person has 2 images with the left light, 2 images with the right light and 2 images with both sides light, so 600 images under three different lighting conditions were used as test images. There are 100 model images and 600 test images in total. The results of the performance with these conditions are presented in Table 2.

Table 2. Performance under varying lighting conditions.

Lighting conditions	Technique	
	MHD	Proposed MHD
Right	62%	75%
Left	60%	74.5%
Both left & right	47%	59.5%

The results in Table 2 indicated that the proposed MHD performed better than the initial MHD by 12.5-14.5 percent and the chosen values of the parameters (W_n, D_n) still suit varying lighting conditions. The lighting condition made a reduction in system performance when we compare it with the accuracy of verifying faces in the ideal condition. When both sides light was used, the edges of the face in both sides had the lower gray-level difference, so the filtering algorithms in LEMExpression.exe software could not detect. Therefore, the error rates were much higher than that of only one light on.

5.3 Face recognition under varying pose of the head

In this subsection, the Bern database was used to compare the performance on face images under the varying pose of the head. Each person has eight face images with eight poses looking to the right, left, upward and downward. These images were used for testing and one frontal face image was used as the model. There are 240 test images and 30 model images in total. The performance of the recognition system based on the proposed MHD is compared with the initial MHD and the results are shown in Table 3.

Table 3. Performance under varying pose of the head.

Pose of the head	Technique	
	MHD	Proposed MHD
Looking to the right	53.33%	51.67%
Looking to the left	46.67%	53.33%
Looking upward	60%	53.33%
Looking downward	56.67%	63.33%

According to our observation, both the initial MHD and the proposed MHD had high error rates under pose different variations. Because the pose changed, the 3D structure of the face was deformed. Therefore, the edge features changed much and the error rate increased.

5.4 Storage and Computational Complexity

Each dominant point has two values need be stored, they are the value of row and column correspond to two 8-bit integers. Therefore, nA is the number of dominant points of the matrix A. The size of the storage requirement of the initial MHD (Mi: bits) is defined as

$$M_i = nA*2*8 \qquad (7)$$

The proposed MHD requires 6 matrices with varying sizes (Algorithm 2). Each element of these matrices corresponds to 1 bit (the value only is 1 or 0). The size of the storage requirement of the proposed MHD (M_p*: bits*) for each image is defined as

$$M_p = (160*160+80*80+40*40+20*20+10*10+5*5) = 34125 \qquad (8)$$

Subsection 3.3 indicates that the computational complexity of computing the directed MHD is $O(nA*nB)$, so the complexity of the initial MHD is $O_i(2*nA*nB)$. These values are the number of the iterations calculated the Euclidean distance while we compute the MHD. Similarly, the complexity of computing the proposed MHD is named O_p.

In this subsection, we used frontal face images from Bern database (BN1 and BN2) and AR database (AR1 and AR2). 30 images BN1 were used as the model images and 30 images BN2 were the test images. Similarly, 100 images AR1 were the model images and 100 images AR2 were the test images. The total number of the dominant points (n), the ratio between storage requirement of the initial MHD and the proposed MHD (M_i/M_p), and the ratio between computational complexity of computing the initial MHD and the proposed MHD (O_i/O_p) were shown in Table 4.

Table 4. Storage requirement and computational complexity of computing.

	BN1	BN2	AR1	AR2
n	12312	12359	29237	28880
M_i / M_p	0.19	0.19	0.14	0.14
O_i / O_p	70.83		50.99	

The results in Table 4 indicated that, compared with the initial MHD, the proposed MHD required over 5-7 times of its storage space. Nevertheless, the computational complexity of the proposed technique was 50-70 times lower than that one of the initial MHD.

6. CONCLUSIONS

The novel technique for computing the MHD, which combines resizing the sizes of the images, classifying the locals and marking the potential ones, has been proposed in this paper. Experiments on frontal faces under ideal condition indicated that the proposed technique for computing the MHD is better than the initial one. The proposed technique achieves 100 percent and 82 percent accuracy of face images in ideal condition on databases Bern and AR, respectively. On other hands, the computational complexity of computing the MHD using proposed technique reduced by 50-70 times compared to the initial MHD. In other experiments with different lighting conditions and varying pose of the head, proposed technique perform highly or equally well as the initial technique. The MHD is a good metric for matching patterns, but it is complex in computing. Decreasing the complexity of computing helps the MHD to become "stronger" in the recognition.

REFERENCES

Muhammad Sharif, Farah Naz, Mussarat Yasmin, Muhammad Alyas Shahid and Amjad Rehman, Face Recognition: A Survey, *Journal of Engineering Science and Technology Review 10 (2)*, pp. 166-177, 2017.

M.-P. Dubuisson and A. K. Jain, "A modified Hausdorff distance for object matching," in *Proceedings of 12th international conference on pattern recognition*, pp. 566–568, 1994.

S. Nutanong, E. H. Jacox, and H. Samet, "An incremental Hausdorff distance calculation algorithm," *Proc. VLDB Endowment*, vol. 4, no. 8, pp. 506–517, May 2011.

A. A. Taha and A. Hanbury, "An efficient algorithm for calculating the exact Hausdorff distance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 11, pp. 2153–2163, Nov. 2015

Y. Chen, F. He, Y. Wu, N. Hou, "A local start search algorithm to compute exact Hausdorff Distance for arbitrary point sets", *Pattern Recognition*, Vol. , No. 67, pp. 139-148, 2017.

D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using the hausdorff distance," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 15, no. 9, pp. 850–863, 1993.

B. Takacs, "Comparing face images using the modified hausdorff distance," *Pattern Recognition*, vol. 31, no. 12, pp. 1873–1881, 1998.

Y. Gao, "Efficiently comparing face images using a modified Hausdorff distance," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 150, no. 6, pp. 346–350, 2003.

Pawana Sharma and Sachin Sharma, "Face recognition and detection using Hausdorff distance, SURF and SVM," *International Journal of Engineering Science and Computing*, vol. 6, no. 3, 2016.

Brunelli, R., Poggio, T., "Face recognition: features versus templates," *IEEE Trans. Pattern Anal. Machine Intell*, 15 (10), pp. 1042–1052, 1993.

Nevatia, R., and Babu, K.R., "Linear feature extraction and description", *Comput. Graph. Image Process.*, 13, pp. 257–269, 1980.

Leung, M.K.H., and Yang, Y.H., "Dynamic two-strip algorithm in curve fitting," *Pattern Recognit.*, 23, pp. 69–79, 1990.

Y. Gao and M. K. Leung, "Face recognition using line edge map," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 6, pp. 764–779, 2002.

Bern Univ. Face Database, ftp://iamftp.unibe.ch/pub/Images/FaceImages/, 2002

A.M. Martinez and R. Benavente, "The AR Face Database," CVC Technical Report no. 24, June 1998.

PHOTOS AND INFORMATION

Bui Thanh Tinh is currently pursuing the degree of telecommunication engineer at Ho Chi Minh University of Technology, Vietnam. His research interests are Computer Vision and Digital Signal Processing.

Truong Thien Nhan is currently pursuing the degree of telecommunication engineer at Ho Chi Minh University of Technology, Vietnam. His research interests are Image Processing and Digital Signal Processing.