

Affine Shape Comparison Using Different Distances

Khalid Aznag[#], Toufik Datsi[#], Ahmed El Oirrak[#], Essaid El Bachari[#]

[#] Computer Science Department, Cadi Ayyad University, Marrakesh, Morocco
E-mail: khalid.aznag@ced.uca.ac.ma

Abstract— In this work, we propose to compare affine shape using Hausdorff distance (HD), Dynamic Time Warping (DTW), Frechet (DF), and Earth Mover distance (EMD). Where there is only a change in resolution shape distance are computed between shape coordinates because the distance is not invariant under rotation or affinity. In case of transformation, distances are calculated not between shape coordinates but between Arc length or Affine Arc length. Arc length is invariant under rotation while Affine Arc length is invariant under affinity. The main advantage is invariance under change of resolution, rotation, and affinity.

Keywords— HD; EMD; DF; DTW; invariant; arc length; affine arc length.

I. INTRODUCTION

A study of affine shape is needed in certain problems that arise in bioinformatics and Pattern Recognition. The affine shape of finite point configurations has been introduced and studied in a series of papers by Sparr [1], [2]. In [2] a reconstruction algorithm, by using affine shape, has been proposed. To retrieve an image from a large database, we need shape descriptors that have sufficient discriminatory properties and are robust to noise. These descriptors should be invariant to translation, rotation, scale and also affinity.

Shape matching is a fundamental problem in computer vision: given two shapes and , one wants to determine how closely resembles , according to some distance measure between the shapes. In order to measure the similarity of and independently of transformations such as translations or rotations and affinity. The problem has received a lot of attention, both in the computer vision and computational geometry community; see the surveys by Hagedoorn and Vetkamp [3] and Alt and Guibas [4]. An efficient and robust shape representation method for shape similarity retrieval is proposed by Abbasi et al [5]. In [6] examine the performance of the representations under affine transformation. The paper [7] introduces a new technique for multi-scale curvature computation on a smoothed 3-D surface.

In this paper, we evaluate each metric in two ways. We first examine the distances produced when the metric is used to discriminate one shape from another. We then examine the metric under transformations and under change of resolution. Figure 1 presents all the possible scenario to compare two shapes.

The remainder of this paper is organized as follows: In Section II, we briefly provide background information on Frechet distance, Earth Mover distance, Dynamic time warping and Harsdorf distance.

Abbreviations	
DTW	Dynamic Time Warping
HD	Hausdorff distance
EMD	Earth Mover distance
DF	Frechet distance
Notation	
X	Curve with 31 points.
\tilde{X}	Transformed curve of X .
X_i	Point on curve X .
\tilde{X}_i	Point on curve \tilde{X} .
X_r	Denote the same curve X but with a change in resolution (curve with 51 points).
X_b	Noised curve.
X_s	Transformed curve of X by similitude (rotation + translation).
X_a	Transformed curve of X by affinity.
L_x	Arc Length for X .
$L_{\tilde{x}}$	Arc Length for \tilde{X} .
AL_x	Affine Arc Length for X .
$AL_{\tilde{x}}$	Affine Arc Length for \tilde{X} .

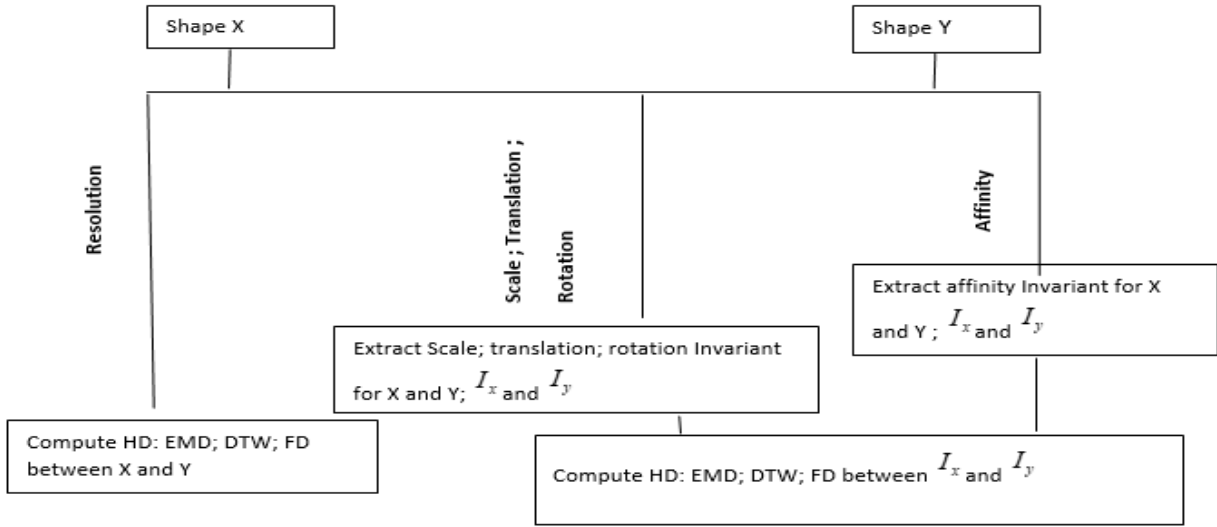


Fig. 1 Scenario to compare two shapes

The remainder of this paper is organized as follows: In Section II, we briefly provide background information on Frechet distance, Earth Mover distance, Dynamic time warping and Harsdorf distance. In Section III, we use Arc length or Affine Arc length to compare between shapes under similitude or affinity transformation, section IV presents some experiments results and discussion. Finally, we conclude the work and highlight some possible perspectives in Section V.

Let $X = (x, y)$ denote a 2D curve then \tilde{X} denote either (X_r, X_b, X_a, X_s) .

II. PRELIMINARIES

A. Frechet distance

The Frechet distance is a measure of similarity between curves that take into account the location and order of the points along the curves. It is named after Maurice Frechet [8].

Let X and \tilde{X} be two given curves.

$$\begin{cases} X = (x(t), y(t)) \\ \tilde{X} = (\tilde{x}(t), \tilde{y}(t)) \end{cases} \quad (1)$$

Then, the Frechet distance between X and \tilde{X} is defined as the infimum over all reparameterizations α and β of $[0,1]$ of the maximum over all $t \in [0,1]$ of the distance between $X(\alpha(t))$ and $\tilde{X}(\beta(t))$.

In mathematical notation, the Frechet distance $F(X, \tilde{X})$ can be expressed as:

$$F(X, \tilde{X}) = \inf_{\alpha, \beta} \max_{t \in [0,1]} d(X(\alpha(t)), \tilde{X}(\beta(t))) \quad (2)$$

where d is a distance function. For two curves X and \tilde{X} of lengths m and n respectively, discrete distance $DF(X, \tilde{X})$ is given by $DF_{m,n}$, where $d_{m,n}$ is obtained

using the following dynamic programming recurrences (Algorithm 1):

Synthetic curve manipulated in these experiments is given by the following equation:

$$\begin{cases} x(t) = 2 \cos(t) \\ y(t) = \sin(t) + 0.5 \sin(5t) \end{cases} \quad \text{where } t \in [0, 2\pi] \quad (3)$$

Algorithm 1 Discrete Frechet distance

```

1: Procedure  $DF(X, \tilde{X})$  // The DF between  $X$  and  $\tilde{X}$ 
2:  $DF_{0,0} \leftarrow d(X_0, \tilde{X}_0)$ 
3: for  $j=1$  to  $n$  do
4:    $DF_{0,j} = \max(DF_{0,j-1}, d(X_0, \tilde{X}_j))$ 
5: end for
6: for  $i=1$  to  $m$  do
7:    $DF_{i,0} = \max(DF_{i-1,0}, d(X_i, \tilde{X}_0))$ 
8: end for
9: for  $i=1$  to  $m$  do
10:  for  $j=1$  to  $n$  do
11:     $DF_{i,j} = \max(\min(DF_{i-1,j}, DF_{i,j-1}, DF_{i-1,j-1}), d(X_i, \tilde{X}_j))$ 
12:  end for
13: end for
14: return  $DF_{m,n}$  // The distance is  $DF_{m,n}$ 
15: end procedure

```

In the experiments, we choose $n=31$ and $m=51$ sizes for original curve and for the same curve but with another resolution.

B. Earth mover distance

The Earth Mover's Distance (EMD) [9] is a method to evaluate dissimilarity between two multidimensional distributions in some feature space where a distance measure between single features, which we call the ground distance is given. This can be formalized as the following linear programming problem:

Let $X = (X_1, w_{x_1}), \dots, (X_m, w_{x_m})$ be the first signature with m clusters, where X_i is the cluster representative and w_{x_i} is the weight of the cluster; and $\tilde{X} = (\tilde{X}_1, w_{\tilde{x}_1}), \dots, (\tilde{X}_n, w_{\tilde{x}_n})$ the second signature with n clusters; and $D = [d_{ij}]$ the ground distance matrix where d_{ij} is the ground distance between clusters X_i and \tilde{X}_j . We want to find a flow $F = [f_{ij}]$, with f_{ij} the flow between X_i and \tilde{X}_j , that minimizes the overall cost subject to the following constraints:

$$W(X, \tilde{X}, F) = \sum_{i=1}^m \sum_{j=1}^n f_{ij} d_{ij} \quad (4)$$

$$\begin{cases} f_{ij} \geq 0, 1 \leq i \leq m, 1 \leq j \leq n \\ \sum_{j=1}^n f_{ij} \leq w_{x_i}, 1 \leq i \leq m \\ \sum_{i=1}^m f_{ij} \leq w_{\tilde{x}_j}, 1 \leq j \leq n \end{cases} \quad (5)$$

$$\sum_{i=1}^m \sum_{j=1}^n f_{ij} = \min\left(\sum_{i=1}^m w_{x_i}, \sum_{j=1}^n w_{\tilde{x}_j}\right) \quad (6)$$

The first constraint allows moving supplies from X to \tilde{X} and not vice versa. The next two constraints limits the amount of supplies that can be sent by the clusters in X to their weights, and the clusters in \tilde{X} to receive no more supplies than their weights; and the last constraint forces to move the maximum amount of supplies possible. We call this amount the total flow. Once the transportation problem is solved, and we have found the optimal flow F^* , the earth mover's distance is defined as the work normalized by the total flow:

$$EMD(X, \tilde{X}) = \frac{\sum_{i=1}^m \sum_{j=1}^n f_{ij}^* d_{ij}}{\sum_{i=1}^m \sum_{j=1}^n f_{ij}^*} \quad (7)$$

C. Dynamic time warping

Dynamic time warping (DTW) [10] is an algorithm for measuring the similarity between two temporal sequences which may vary in time or speed. The following code (Algorithm 2) illustrates the implementation of the dynamic time warping algorithm.

Algorithm 2 Dynamic Time Warping Algorithm

```

1: Procedure  $DTW(X, \tilde{X})$  // The DTW between  $X$  and  $\tilde{X}$ 
2:  $DTW_{0,0} \leftarrow 0$ 
3: for  $i=1$  to  $n$  do
4:    $DTW_{i,0} \leftarrow \infty$ 
5: end for
6: for  $j=1$  to  $m$  do
7:    $DTW_{0,j} \leftarrow \infty$ 
```

```

8: end for
9: for  $i=1$  to  $m$  do
10:  for  $j=1$  to  $n$  do
11:     $cost \leftarrow d(X_i, \tilde{X}_j)$ 
12:     $DTW_{ij} \leftarrow cost + \min(DTW_{i-1,j}, DTW_{i,j-1}, DTW_{i-1,j-1})$ 
13:  end for
14: return  $DTW_{mn}$  // The distance is  $DTW_{mn}$ 
15: end procedure
```

D. Hausdorff distance

The Hausdorff distance [11] is a mathematical construct to measure the "closeness" of two sets of points that are subsets of a metric space.

Let X and \tilde{X} be two 2D curves. We define their Hausdorff distance $DH(X, \tilde{X})$ by:

$$DH(X, \tilde{X}) = \max \left\{ \sup_{\tilde{X}_j \in \tilde{X}} \inf_{X_i \in X} d(X_i, \tilde{X}_j), \sup_{X_i \in X} \inf_{\tilde{X}_j \in \tilde{X}} d(X_i, \tilde{X}_j) \right\} \quad (8)$$

where *sup* represents the supremum and *inf* the infimum.

E. Chamfer distance

Given a neighbourhood and associated move lengths, the chamfer distance [12] between X and \tilde{X} relative to this neighbourhood is the length of the shortest digital arc form X and \tilde{X} .

$$Ch(X, \tilde{X}) = \sum_{X_i \in X} \min_{\tilde{X}_j \in \tilde{X}} \|X_i - \tilde{X}_j\| \quad (9)$$

III. SIMILITUDE AND AFFINITY TRANSFORMATIONS

In case there is no transform between X and \tilde{X} we can use directly HD, EMD, DF, and DTW. In case there is a transformation we must first extract invariant signatures against this transformation, then we compare these signatures. Among these signatures, we can use arc length in case rotation, and affine arc Length in affinity case.

A. Similitude and Arc length invariance

Curvature: Let X be a smooth curve with position vector $\vec{r}(s)$ where s is the arc length parameter. The curvature k of X is defined by:

$$k = \left\| \frac{d\vec{T}}{ds} \right\| \quad (10)$$

where \vec{T} is the unit tangent vector. If a curve X is defined parametrically by $x(t)$ and $y(t)$, then its arc length is defined by

$$\tau = \int (\dot{x}^2 + \dot{y}^2)^{1/2} \quad (11)$$

Where \dot{x} denote first derivative of x and \dot{y} first derivative of y . Under the action of rotation contours coordinates of the second image (Figure 7) are related to coordinates contours of the first image by

$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \end{pmatrix} = \begin{pmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (12)$$

Then

$$\begin{cases} \tilde{x} = x \cos \phi + y \sin \phi \\ \tilde{y} = -x \sin \phi + y \cos \phi \end{cases} \quad (13)$$

$$\begin{cases} \dot{\tilde{x}} = \dot{x} \cos \phi + \dot{y} \sin \phi \\ \dot{\tilde{y}} = -\dot{x} \sin \phi + \dot{y} \cos \phi \end{cases} \quad (14)$$

So

$$\begin{aligned} \tau &= \int ((\dot{x} \cos \phi + \dot{y} \sin \phi)^2 + (\dot{y} \cos \phi - \dot{x} \sin \phi)^2)^{1/2} \\ &= \int ((\dot{x} \cos \phi)^2 + 2\dot{x}\dot{y} \cos \phi \sin \phi + (\dot{y} \cos \phi)^2 \\ &\quad - 2(\dot{x}\dot{y} \cos \phi \sin \phi + (-\dot{x} \sin \phi)^2)^{1/2} \\ &= \int (\dot{x}^2 + \dot{y}^2)^{1/2} \end{aligned}$$

With

$$\cos^2 \phi + \sin^2 \phi = 1 \quad (15)$$

B. Affinity and affine arc length invariance

Arc Length is invariant under similarity transforms; i.e. rotation, translation, and uniform scaling. However, under general affine transforms, the change in curvature is not a linear function of the transformation matrix. Affine Arc Length has been defined as an alternative for Arc Length which changes linearly under affine transforms. Suppose that X is parameterized curve. Then the special affine curvature (or equi-affine curvature) is given by

$$\int (\dot{x}\ddot{y} - \ddot{x}\dot{y}) \quad (16)$$

Here \dot{x} denotes the first derivative of x and \ddot{x} denote second derivative. The main disadvantage of the affine length is that its computation requires higher order derivatives. The computation starts from convolving each coordinate $x(t)$ and $y(t)$ of the curve with a Gaussian function $g(t, \sigma)$. In continuous form we have:

$$\begin{cases} x(t, \sigma) = x(t) \otimes g(t, \sigma) \\ y(t, \sigma) = y(t) \otimes g(t, \sigma) \end{cases} \quad (17)$$

$$\begin{cases} \dot{x} = x \otimes \dot{g} \\ \dot{y} = y \otimes \dot{g} \end{cases} \quad (18)$$

$$\begin{cases} \ddot{x} = x \otimes \ddot{g} \\ \ddot{y} = y \otimes \ddot{g} \end{cases} \quad (19)$$

Where \otimes denote convolution operator. \dot{x} , \ddot{x} denote respectively first and second derivative approximation. The 1D Gaussian function $g(t, \sigma)$ of width σ is defined as follow:

$$g(t, \sigma) = \frac{1}{\sqrt{(2\pi)}} e^{-\frac{t^2}{2\sigma^2}} \quad (20)$$

Under general affinity coordinated in the two images are related by the formula:

$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \end{pmatrix} = A \begin{pmatrix} x \\ y \end{pmatrix} \quad (21)$$

where

$$A = \begin{pmatrix} a_{12} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \quad (22)$$

The equation (16) is invariant under the action of affinity (with the condition $|A| = 1$) where $|A|$ denote determinant of matrix A . In fact

$$\begin{cases} \tilde{x} = a_{11}x + a_{12}y \\ \tilde{y} = a_{21}x + a_{22}y \end{cases} \quad (23)$$

So

$$\begin{aligned} &(\tilde{x})(\tilde{y}) - (\tilde{x})(\tilde{y}) \\ &= (a_{11}\dot{x} + a_{12}\dot{y})(a_{21}\ddot{x} + a_{22}\ddot{y}) - (a_{11}\ddot{x} + a_{12}\ddot{y})(a_{21}\dot{x} + a_{22}\dot{y}) \\ &= a_{11}a_{21}\dot{x}\ddot{x} + a_{11}a_{22}\dot{x}\ddot{y} + a_{12}a_{21}\dot{y}\ddot{x} + a_{12}a_{22}\dot{y}\ddot{y} - a_{11}a_{21}\ddot{x}\dot{x} - a_{11}a_{22}\ddot{x}\dot{y} - a_{12}a_{21}\ddot{y}\dot{x} - a_{12}a_{22}\ddot{y}\dot{y} \\ &= a_{11}a_{22}\dot{x}\ddot{y} + a_{12}a_{21}\dot{y}\ddot{x} - a_{11}a_{22}\ddot{x}\dot{y} - a_{12}a_{21}\ddot{y}\dot{x} \\ &= a_{12}a_{21}(\dot{x}\ddot{y} - \ddot{x}\dot{y}) + a_{11}a_{22}(\dot{y}\ddot{x} - \ddot{y}\dot{x}) \end{aligned}$$

IV. EXPERIMENTAL RESULTS AND DISCUSSION

In order to demonstrate the effectiveness of the proposed methods, we use the different distances HD, EMD, DTW, and DTW. However, distances EMD, HD, DF and DTW depends on the starting point, for instance, two visually similar curves but a shift can have a very large distance (see Figures 8 and 9). Figure 8 shows the shift between Arc Length, coordinates in case of resolution. It is clear from Table 1 and Table 4 that DF gives good results then DTW doesn't.

A. Curve with resolution and noise

If no transformations are possible between X and \tilde{X} , then we can use directly HD, EMD, DF, and DTW between coordinates; if there is only change in resolution for the shape we compute a distance between shape coordinates because the distance is not invariant under transformation e.g. rotation. As shown in Figure 2; the curve 2.b) is the same as 2.a) but with another resolution, 2.c) noisy curve and 2.d) affinity curve. Tables 1, 2, 3 and 4 present some results.

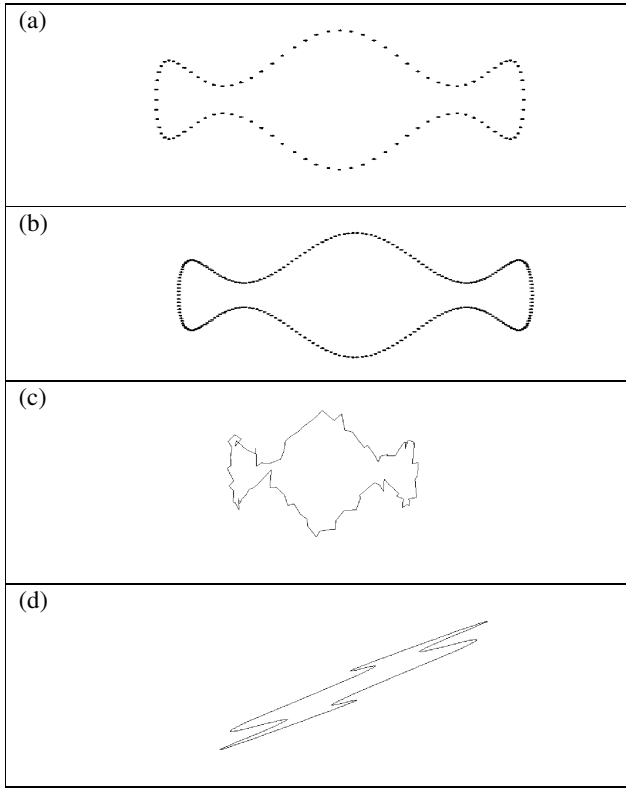


Fig. 2 (a). Test curve with resolution 30, (b). Test curve with resolution 50, (c). Test curve with noise added, (d). Affinity test shape

TABLE I

DF BETWEEN X, Y, X_b AND X_r

$DF(X, X_r)$	$DF(X, Y)$	$DF(X, X_b)$
0.2317	39.4911	0.1622

TABLE II

EMD BETWEEN X, Y, X_b AND X_r

$EMD(X, X_r)$	$EMD(X, Y)$	$EMD(X, X_b)$
0.0484	5.3797	0.0825

TABLE III

HD BETWEEN X, Y, X_b AND X_r

$HD(X, X_r)$	$HD(X, Y)$	$HD(X, X_b)$
0.2317	38.0591	0.3297

TABLE IV

DTW BETWEEN X, Y, X_b AND X_r

$DTW(X, X_r)$	$DTW(X, Y)$	$DTW(X, X_b)$
5.1880	654.6982	3.3882

B. Arc length invariant under rotation

In case there is a transformation we must extract invariant signature against the transformations, then we compare these signatures, for example for our case: a rotation we propose first to extract a quantity invariant with respect to rotation, then secondly we calculate HD, EMD,

DF between these invariant, among these signatures we can use Arc Length, and Affine Arc Length. Also, in general if there is transformation distances are calculated not between shape coordinates but between Arc length. Figure 3 shows a curve and its transformed (rotation). Figure 4 shows the Arc length for curves in Figure 3. Table 5 presents a different distance between the curve and its transformed. In Table 6 different distance between Arc length are presented.

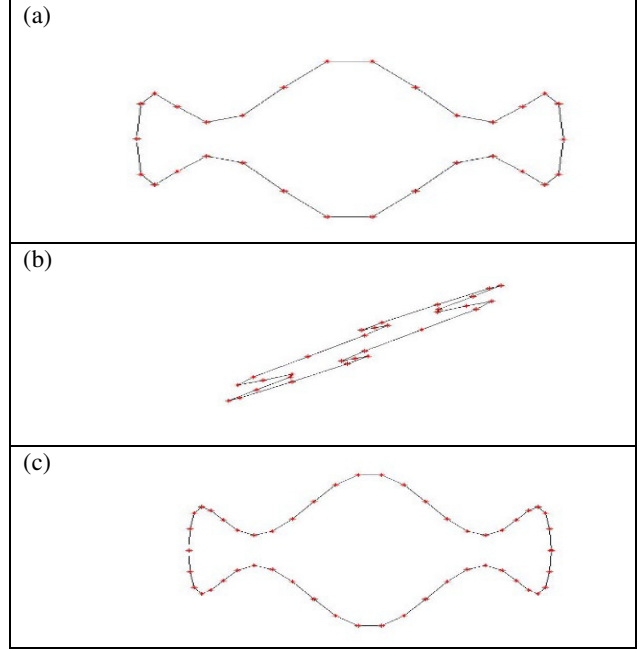


Fig. 3 Curves and its transformed. (a) Curve. (b) Rotation. (c) Resolution

TABLE V

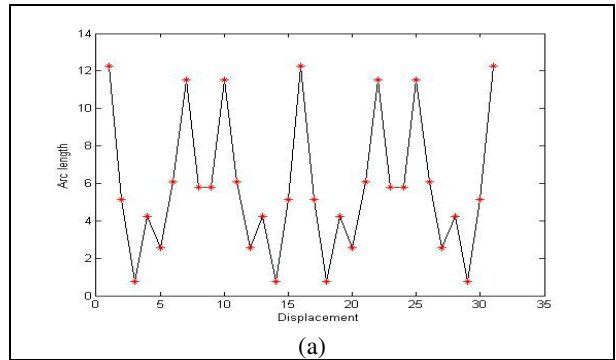
DIFFERENT DISTANCE BETWEEN ARC LENGTH

Distance	$D(L_X, L_{\tilde{X}})$
HD	420.0281
DTW	3.2411e + 003
Chamfer	54.5073

TABLE VI

DISTANCE BETWEEN ARC LENGTH

$HD(X, X_a)$	$DF(X, X_a)$	$EMD(X, X_a)$
311.5896	311.5896	78.5955



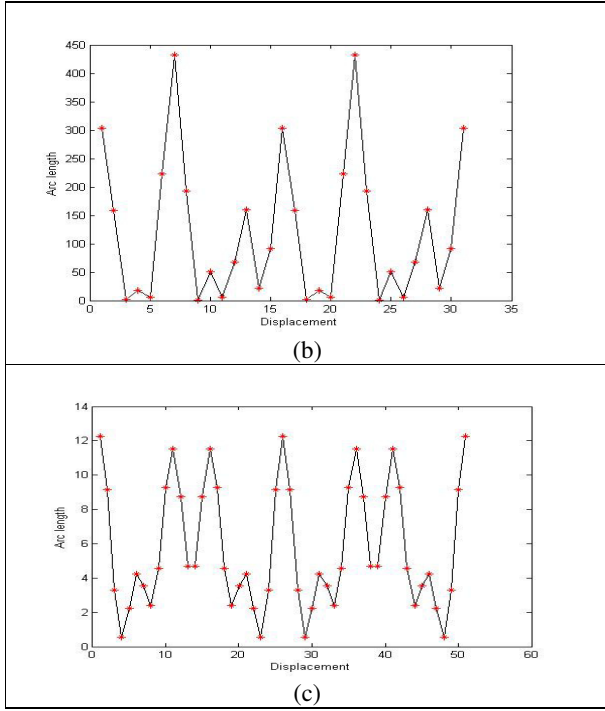


Fig. 4 Arc length for curves in Figure 3; a, b and c respectively

C. Affine Arc length under affinity

In case the affinity distance is calculated between Affine Arc length, Affine Arc length is invariant. Affinity matrix used in our case is defined by:

$$A = \begin{pmatrix} -0.3265 & 0.3852 \\ -5.0838 & 2.9351 \end{pmatrix} \quad (24)$$

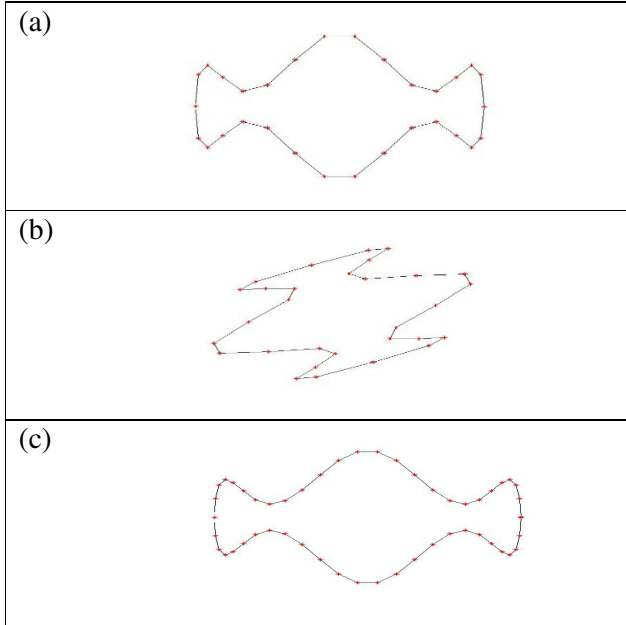


Fig. 5 Curves and its transformed. (a) Curve. (b) Transformed rotation. (c) Resolution

Figure 5 shows a curve and its transformed (affinity). Figure 6 shows the Affine Arc length for curves in Figure 5. In Table 7 different distance between the curve and its

transform are presented. In Table 8 different distance between Affine Arc length are presented.

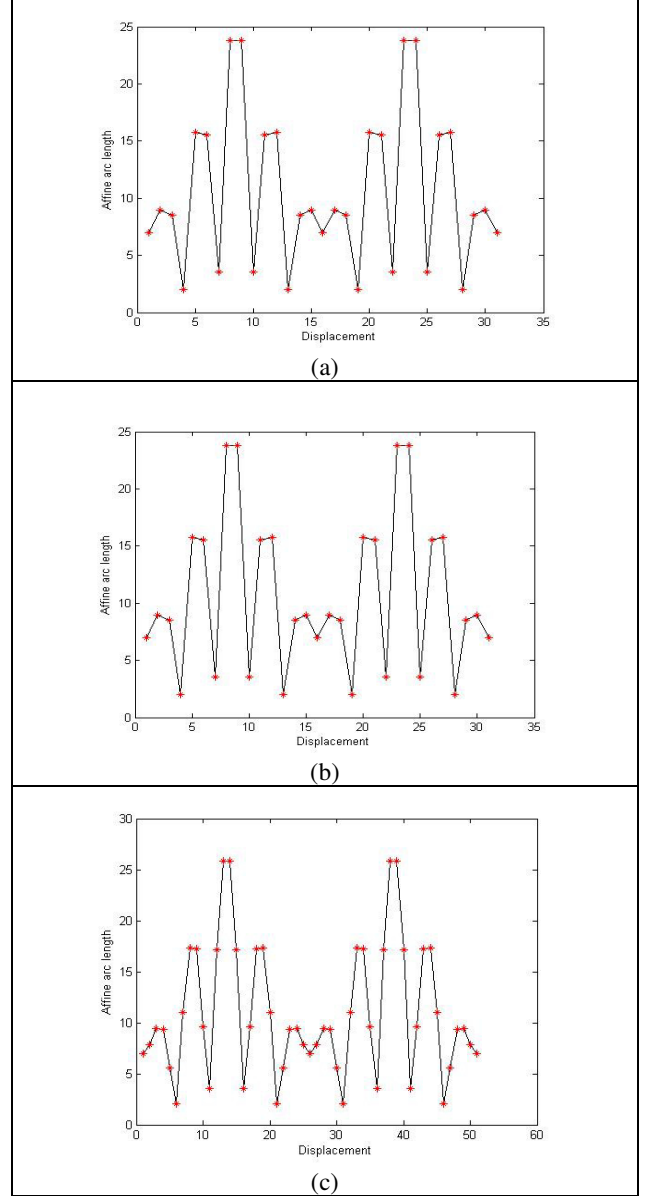


Fig. 6 Affine Arc Length for curves in Figure 5; a, b and c respectively

TABLE VII
DISTANCE BETWEEN AFFINE ARC LENGTH

$HD(X, X_a)$	$DF(X, X_a)$	$EMD(X, X_a)$
1.5987×10^{-14}	2.3093×10^{-14}	3.3750×10^{-13}

TABLE VIII
DIFFERENT DISTANCE BETWEEN AFFINITY ARC LENGTH

Distance	$D(X, X_a)$
HD	2.0611
DTW	108.3047
Chamfer	24.7682

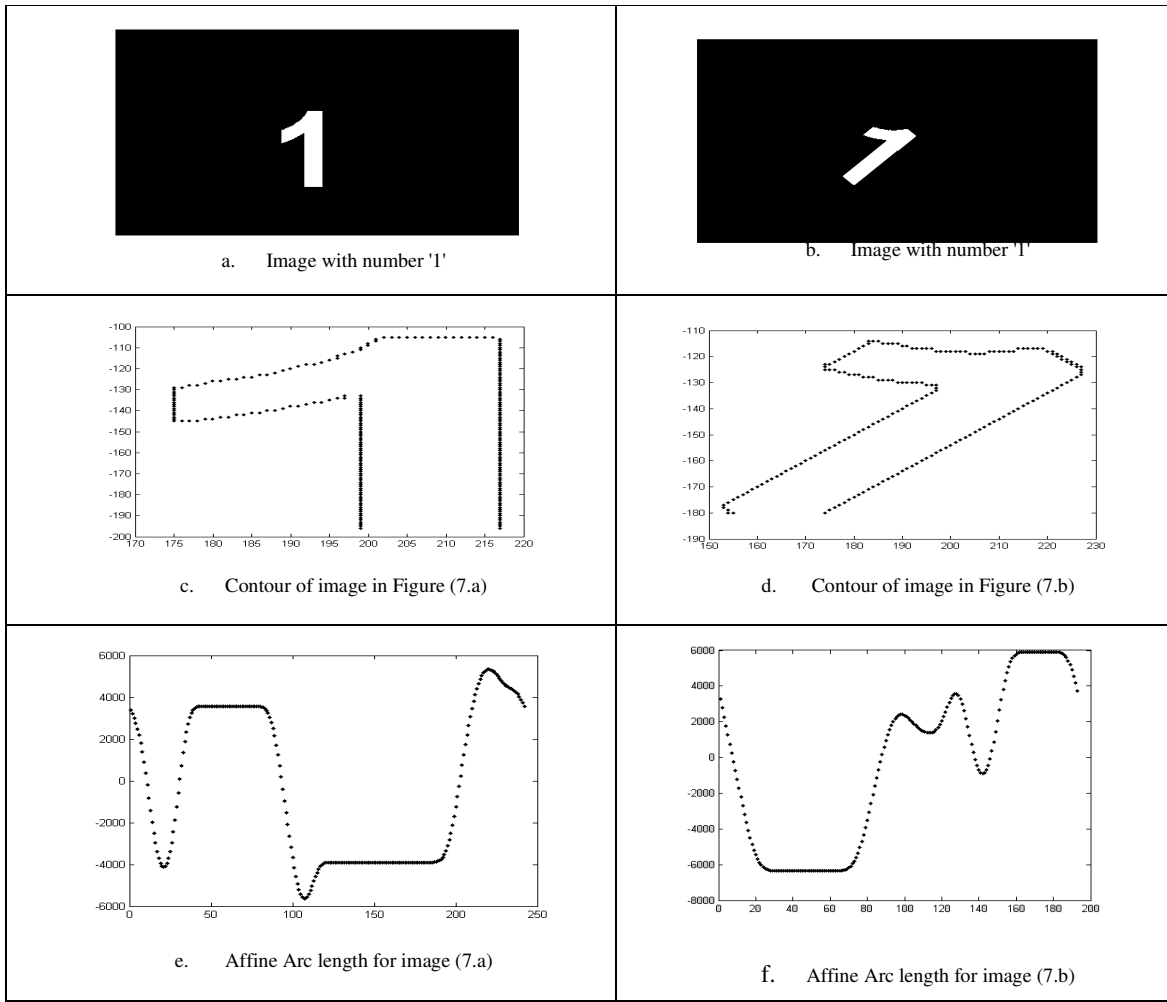


Fig. 7 Affine Arc Length for real image

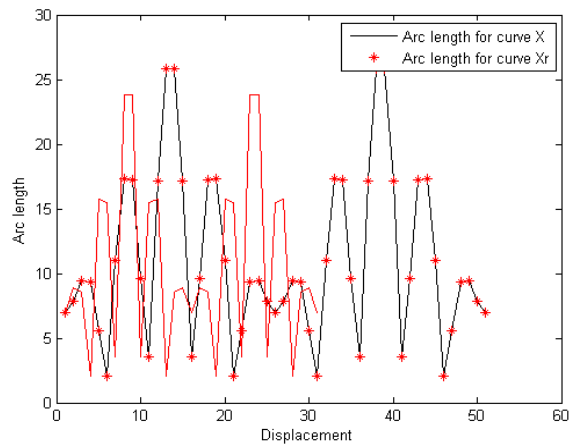


Fig. 8 Arc Length for X and X_r

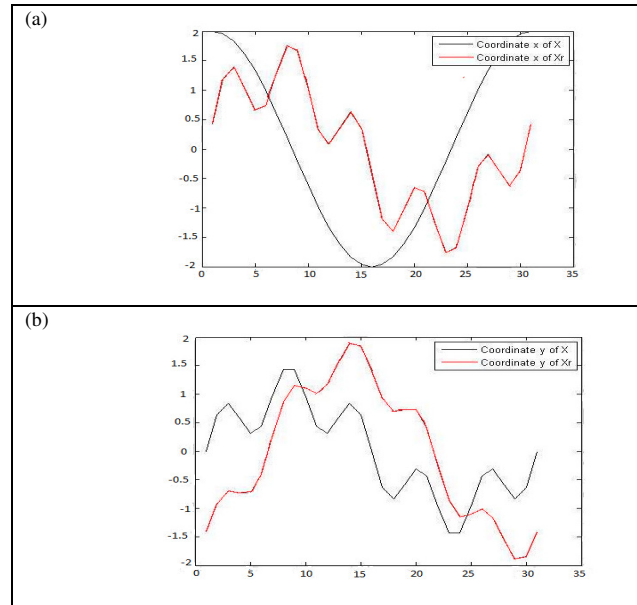


Fig. 9 (a):Coordinates x of X and X_r (b):Coordinates y of X and X_r

V. CONCLUSIONS AND FUTURE WORKS

In this paper we have presented a comparison affine shape with different types of distances. We showed that DTW, HD, EMD, and DF are not invariant under rotation and affinity. If no transformation is possible between curves, we can use directly HD, EMD, DF, and DTW. In case there is a transformation we extract invariant signature, then we compare these signatures. But using Arc length and Affine Arc length invariance is assured respectively for rotation and affinity. Where there is shift DTW is not suitable. As future perspectives, we are working on integrating our study in the context of a 3D object.

REFERENCES

- [1] G. Sparr, "A common framework for kinetic depth, reconstruction and motion for deformable objects," In ECCV (2) , 1994, pp. 471-482.
- [2] G. Sparr, "Structure and motion from kinetic depth," In: The Sophus Lie International Workshop on Computer Vision and Applied Geometry, 1995, Vol. 2.
- [3] O. Klein, and R. C. Veltkamp, "Approximation algorithms for computing the earth mover's distance under transformations," In International Symposium on Algorithms and Computation. Springer, Berlin, Heidelberg, pp. 1019-1028, 2005.
- [4] H. Alt, and L.J. Guibas, "Discrete geometric shapes: Matching, interpolation, and approximation," In Handbook of computational geometry, 2000, pp. 121-153.
- [5] S. Abbasi, F. Mokhtarian, and J. Kittler, "Curvature scale space image in shape similarity retrieval," Multimedia systems, 1999, Vol. 7, No. 6, pp. 467-476.
- [6] F. Mokhtarian, and S. Abbasi, "Curvature scale space for shape similarity retrieval under affine transforms," In International Conference on Computer Analysis of Images and Patterns. Springer, Berlin, Heidelberg, 1999, pp. 65-72.
- [7] P. Yuen, N. Khalili, and F. Mokhtarian, "Curvature Estimation on Smoothed 3-D Meshes ," In BMVC, 1999, pp. 1-10.
- [8] T. Wylie, and B. Zhu, "Following a curve with the discrete Fréchet distance," Theoretical Computer Science, 2014, Vol. 556, pp. 34-44.
- [9] M. Zhang, Y. Liu, H. Luan, and M. Sun, "Earth mover's distance minimization for unsupervised bilingual lexicon induction," In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 1934-1945.
- [10] Z. Zhang, R. Tavenard, A. Bailly, X. Tang, P. Tang, and T. Corpetti , "Dynamic time warping under limited warping path length," Information Sciences, 2017, 393, 91-107.
- [11] X.D. Chen, W. Ma, G. Xu, and J. C. Paul, "Computing the Hausdorff distance between two B-spline curves," Computer-Aided Design, 2010, Vol. 42, No. 12, pp.1197-1206.
- [12] L. Dantanarayana, G. Dissanayake and R. Ranasinge, "C-log: A chamfer distance-based algorithm for localisation in occupancy grid-maps," CAAI transactions on intelligence technology, 2016, 1(3), 272-284.