

IMAGE SIMILARITY USING SYMBOLIC REPRESENTATION AND ITS VARIATIONS

Hala Ahmed Abdul-Moneim

Dept. of Mathematics, Faculty of Science, Minia University

ABSTRACT

This paper proposes a new method for image/object retrieval. A pre-processing technique is applied to describe the object, in one dimensional representation, as a pseudo time series. The proposed algorithm develops the modified versions of the SAX representation: applies an approach called Extended SAX (ESAX) in order to realize efficient and accurate discovering of important patterns, necessary for retrieving the most plausible similar objects. Our approach depends upon a table contains the break-points that divide a Gaussian distribution in an arbitrary number of equiprobable regions. Each breakpoint has more than one cardinality. A distance measure is used to decide the most plausible matching between strings of symbolic words. The experimental results have shown that our approach improves detection accuracy.

KEYWORDS

1-D representation of objects, Symbolic Aggregate approXimation(SAX), Shape number, Time series, Extended SAX representation.

1. INTRODUCTION

Since the one-dimensional representations have been shown to achieve comparable or superior accuracy in object/image matching, we identify the object and reduce the representation to a 1-D representation by describing the object as a pseudo time series. Normally, dealing with time series is difficult, mainly, due to its very high dimensionality. Our work is interested in finding the most plausible matching, which aims to achieve cardinality of the analyzed object and reducing the representation to a 1-D representation. We apply an approach called Extended SAX (ESAX) which is based on the SAX representation introduced by Lin, Keogh, Lonardi and Chiu [1]. A SAX representation of time series consists of compact sequences of symbols (say, alphabets) that can be efficiently compared and consequently speeds up the retrieval task.

The rest of the paper is organized as follows: Section 2 presents an overview of the symbolic method. representations. Section 3 presents the modified SAX representations. Section 4 describes the proposed technique Section 5 presents the matching procedure. In section 6 we describe the similarity measurement. The results of applying the proposed techniques on some image datasets are presented in section 7. Section 8 gives the conclusion.

2. PREVIOUS WORK

Object descriptors techniques of two types: (A) Descriptors based on external space domain techniques which utilize boundary primitives. These techniques are generally efficient because it preserves information and allows considerable data reduction. Two of these techniques are widely used nowadays, the Freeman boundary Chain Code technique and the technique of encoding the distance from the shape centroid to the shape boundary. (B) Descriptors based on internal space domain techniques which utilize structural or relational properties derived from the complete shape (shape region).

We intent to provide Some useful popular simple techniques of the external space domain techniques type.

Zhang and Lu. [2] Common methods of boundary decomposition are based on polygonal approximation, curvature decomposition and curve fitting.

The set of predefined primitives is called the codebook and the primitives are called codewords. For example, given the codewords in the right of Figure 1, the chromosome shape in the left of Fig. 1 can be represented as a grammatical string of S where $S = dbabcbabdbabcbab$, then the matching between shapes can use string matching .

Berretti, Bimbo and Pala [3] extended the model used in [4] for general shape retrieval. Fig. 2 shows a horse shape has been divided into different ‘tokens’. Since the feature includes curve orientation, it is not rotation invariant.

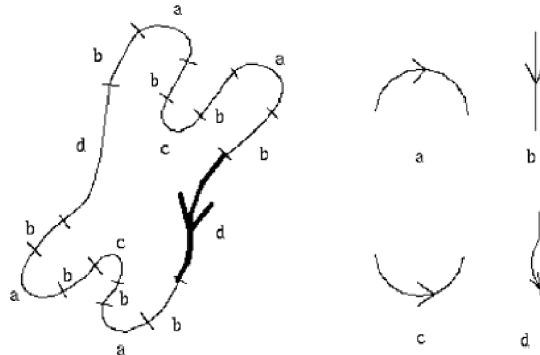


Figure 1. Structural description of chromosome shape (reprinted from [2]).

Given a query object, the retrieval of similar objects from the database takes two steps. The first step is token retrieval. The set of retrieved tokens having the same shape identifier form a potential similar shape.

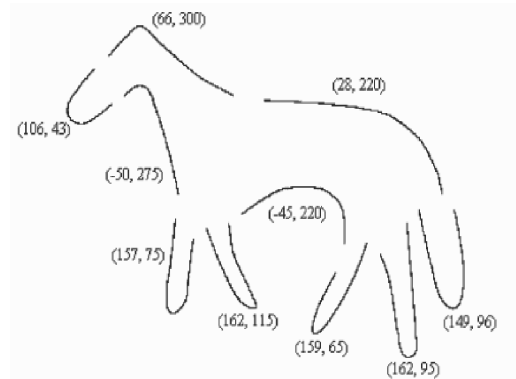


Figure 2. A horse shape has been divided into different 'tokens' (reprinted from [2]).

Chain code describes an object by a sequence of unit-size line segments with a given orientation. The method was introduced in 1961 by Freeman [5] who described a method permitting the encoding of arbitrary geometric configurations. This code follows the boundary in counter clockwise manner and keeps track of the direction as we go from one contour pixel to the next. The codes involve 4-connected and 8-connected paths as shown in Figure 3(a) and Figure 3(b). For processing an image to become a chain code, it is common to have binary image as input. This allows figure separation by thresholding.

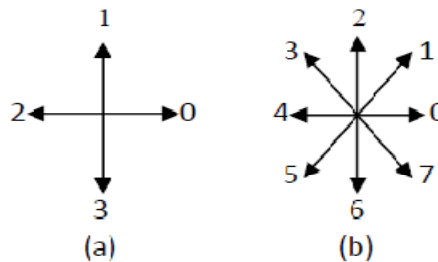


Figure 3. Neighbour Directions of Freeman chain code(FCC).

One of the important things before having the chain code of an image is edge detection process. It is often used as an input to a higher level analysis. For example, it can be used for polygon approximation and for finding boundary curvature which is a important perceptual feature.

To know about the arithmetic coding used to compress images, see [6].

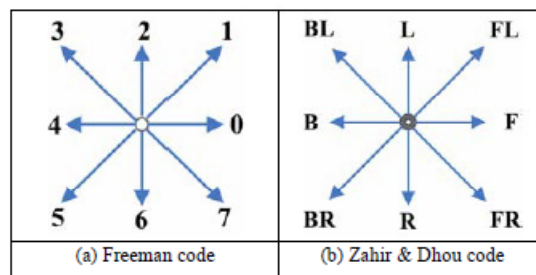


Figure 4 (a) Freeman chain code and (b) Z&D Code

Xi, Keogh, Wei and Mafrá-Neto [7] determined the similarity between two shapes by comparing their Symbolic Aggregate Approximation (SAX) representation. This 1-D representation has the advantage of being simple, completely parameter free and there is no effects of scale within the image. Figure 5 shows an example of the process of converting an image into pseudo time series. There are dozens of techniques in the literature for converting shapes to time series [8] like the technique “column-wise/row-wise Aggregate Approximation (CAA/RAA) technique”, [9] which we are applying in this paper.

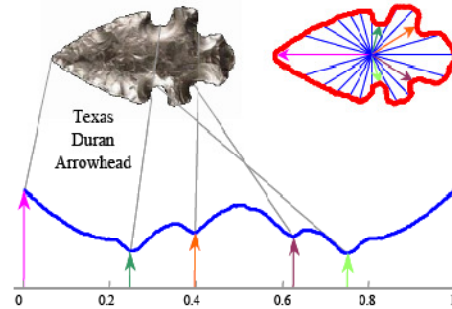


Figure 5. Shapes can be converted to time series.

Shieh and Keogh [10] introduced the iSAX technique to allow extensible hashing. The SAX word shown in Figure 6 (iv) can be written as $SAX(T,4,2) = T_2 = \{1, 1, 0, 0\}$.

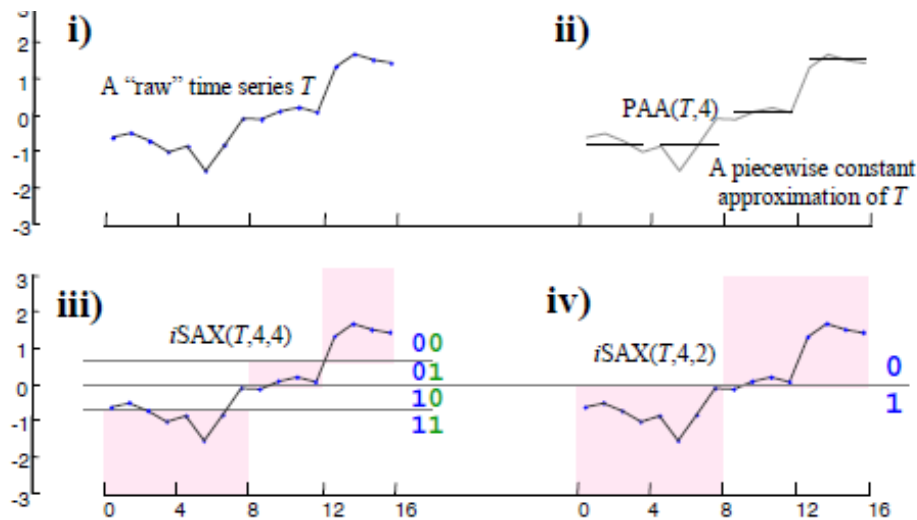


Figure 6. i) A time series T , of length 16. ii) A PAA approximation of T , with 4 segments. A time series T converted into SAX words of cardinality 4 (iii), and cardinality 2 (iv).

3. PRELIMINARIES

Definition 1: Time Sequence/series

A time sequence/series is a sequence (ordered collection) of real values

Definition 2: Subsequence

Figure 7 (b) shows a subsequence of the time sequence shown in Figure 7 (a).

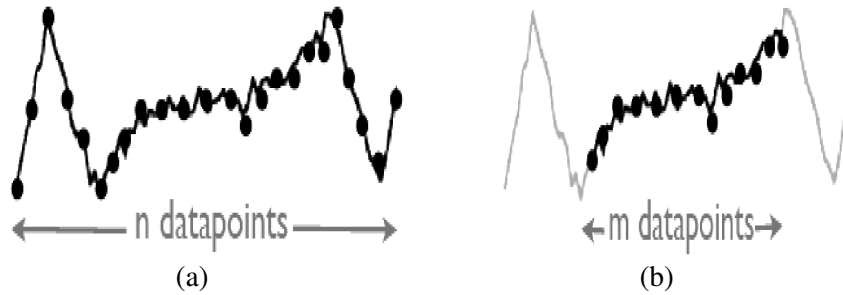


Figure 7. (a) A sequence and (b) a subsequence [11].

Definition 3: Piecewise Aggregate Approximation

Piecewise Aggregate Approximation can be defined as a dimensionality reduction. Keogh and Pazzani [12] reduced the data from n dimensions to N dimensions by dividing the data into N equi-sized "frames". The mean value of the data falling within a frame is calculated and a vector of these values becomes the data reduced representation. Figure 8 shows an illustration of this data reduction technique.

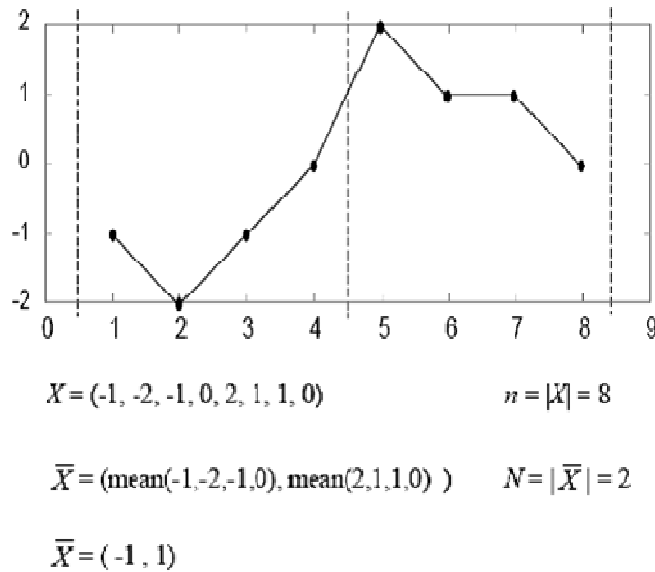


Figure 8. A time series consisting of eight ($n=8$) points is projected into two ($N=2$) dimensions.

Definition 4: Breakpoints

These breakpoints may be determined by referring to a statistical table. For example, Table 1 gives the breakpoints for values of p from 3 to 10.

Table 1.

$\beta_i \backslash a$	3	4	5	6	7	8	9	10	
β_1	-0.43	-0.67	-0.84	-0.97	-1.07	-1.15	-1.22	-1.28	a
β_2	0.43	0	-0.25	-0.43	-0.57	-0.67	-0.76	-0.84	b
β_3		0.67	0.25	0	-0.18	-0.32	-0.43	-0.52	c
β_4			0.84	0.43	0.18	0	-0.14	-0.25	d
β_5				0.97	0.57	0.32	0.14	0	e
β_6					1.07	0.67	0.43	0.25	f
β_7						1.15	0.76	0.52	g
β_8							1.22	0.84	h
β_9								1.28	i
									j

4. PROBLEM DEFINITION

Given: a query compact sequence of symbols (say, alphabets sequence q), a database S of N sequences S_1, S_2, \dots, S_N , (N shapes), a distance measure D and a tolerance threshold ϵ , find: the set of sequences R (match with q) in S that are within distance ϵ from q .

5. SYMBOLIZING THE PSEUDO "TIME SERIES"

Symbolization serves several important purposes. It provides us with a lower dimensional representation that reduces the noise effect in the raw time series while preserving its properties[13, 14]. Figure 9 shows the SAX technique. This is easily achieved since normalized time series have highly Gaussian distributions, Lin., Keogh, Lonardi, and Chiu [13].

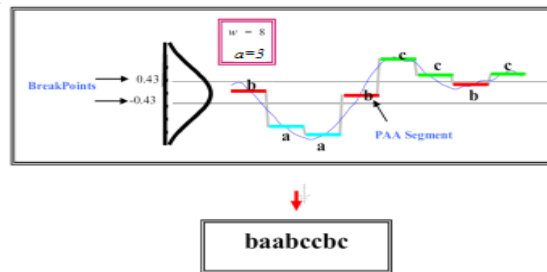


Figure 9. A time series (smooth curve) is converted to Piecewise Aggregate Approximation (PAA) segments and then using the pre-determined breakpoints the PAA coefficients are mapped into symbols (SAX representation)[16].

The following formula is used to normalize the time series: $TS = \frac{(TS - \text{mean}(TS))}{\text{std}(TS)}$, where TS is the time series, $\text{mean}(TS)$ is the mean value of time series variables, and $\text{std}(TS)$ is the standard deviation of the time series variables.

After mapping all coefficients to their corresponding symbols, we get an ESAX representation of time series.

As an example, the steps of getting the symbolic representation of the time series obtained by using the CAA/RAA technique are as follows :

Step 1:

We normalize the image/shape before computing CAA/RAA, by resizing it to be 128x192, as in [11], so that each image has the same number of rows (192) and the same number of columns (128). Thus, the CAA/RAA for any image will be of fixed length.

Step 2:

Compute the CAA/RAA: After aligning the major axis of the object with the x-axis, as shown in Figure 10, the mean value of each column/row is computed. Figure 11 shows the plots of the 1-D representations (CAAs and RAAs) of some objects.

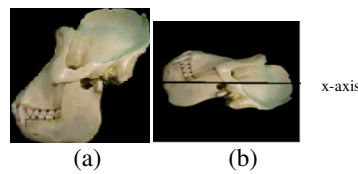


Figure 10. (a) Origin image, (b) Align the major axis with x-axis.

Step 3:

Table 1 gives the breakpoints values for the size of the alphabet from 3 to 10. A numerical example of the SAX technique is shown in Tables 2 and 3. Table 2 shows the raw time series which was divided into 16 segments. Each segment contains 8 datapoints. Figure 12 shows the ESAX representation of a given time series where each segment is represented by a sequence of three alphabet symbols. Table 3 shows the ESAX representation of each segment in Table 2 for $a=2:a=10$.

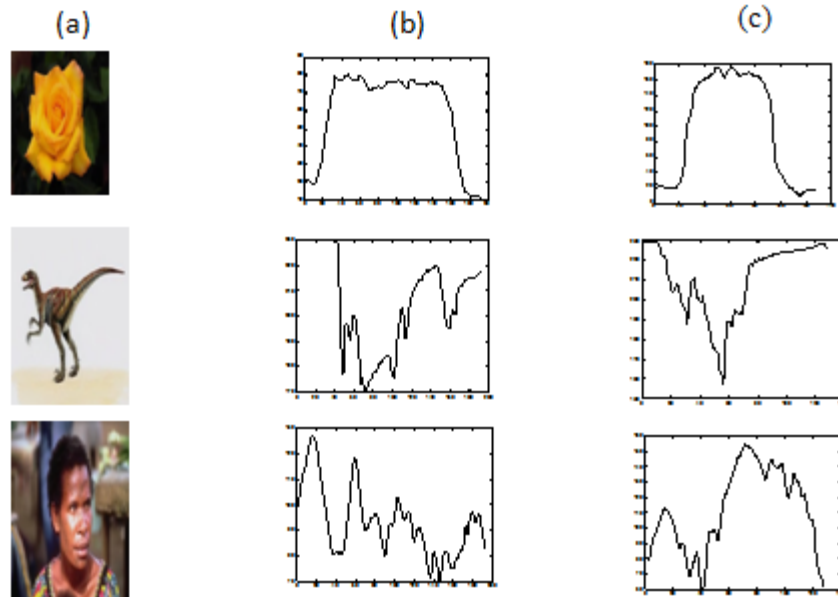


Figure 11. (a)The original image, (b) the plot of its CAA, and (c) the plot of its RAA.

Step 4:

After obtaining the PAA of the pseudo time series , we have the equal sized segments and its mean values. Any value of the three values-based segment that falls in that region can then be mapped to the appropriate label. An ESAX word is simply then a vector of discrete alphabet of symbols with cardinality a . Table 3 shows the ESAX word of 16 segments with cardinality a from 2 :10 (Alphabet size). For example, the ESAX word of segment12 can be written as {hge} with cardinality $a =10$ (Alphabet size= a,b,c,d,e,f,g,h,i,j). We denote this word-segment as w_{12}^{10} , and assume that it is produced by the function $ESAX(s_1,16,10)$. The “w” is written in boldface to distinguish it from the Alphabet from which it was derived. The superscript of “10” denotes the cardinality of the symbols and the subscript denotes the segment number.

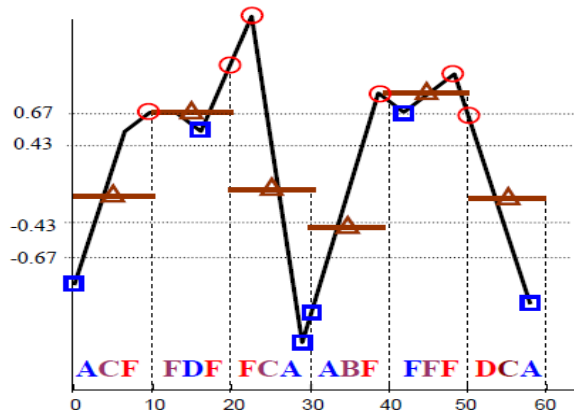


Figure 12: The Extended SAX representation is ACFFDFFFCAABFFFFDCA.

Table 2 : The raw time series was divided into 16 segments. Each segment contains 8 datapoints.

Deviding the 1-D representation of 128 datapoints (RAA/CAA) to 16 segments then each segment will have 8 datapoints.								
Segment1	0.57725	0.56726	0.55243	0.5365	0.51991	0.49775	0.48172	0.46121
Segment2	0	0	0	0	0	0	0	0
Segment3	0.66219	0.65362	0.64318	0.63515	0.62384	0.61485	0.60386	0.59161
Segment4	0.79594	0.79671	0.79761	0.79383	0.79205	0.79498	0.78989	0.77298
Segment5	0.71867	0.71399	0.69931	0.69375	0.68924	0.68241	0.6759	0.67096
Segment6	0.43807	0.41295	0.38672	0.35821	0.32543	0.28627	0.24056	0.17996
Segment7	0.54353	0.55686	0.56677	0.58284	0.59183	0.60607	0.61603	0.62698
Segment8	0.75807	0.76444	0.77251	0.77604	0.78321	0.78595	0.78546	0.79414
Segment9	0	0	0.75807	0	0.00053105	0.54353	0.66219	0.43807
Segment10	0.73544	0	0	0.41295	0.76444	0	0.38	0.65362
Segment11	0.00022467	0.55686	0.7373	0	0	0.38672	0.40531	0.64318
Segment12	0.77251	0	0.07886	0.56677	0.73658	0.42855	0.07886	0.47279
Segment13	0.63515	0.35821	0.77604	0	0.17623	0.58284	0.38672	0.60607
Segment14	0.7335	0.45188	0.62384	0.32543	0.78321	0	0.66219	0.48899
Segment15	0.17996	0.28627	0.23135	0.59183	0.72868	0.47279	0.61485	0.28627
Segment16	0.52792	0.48899	0.78595	0	0.2788	0.60607	0.7251	0.48899

Table 3 : ESAX representation of each segment in Table 2 for a=3 : a=10.

Segment number	max	mean	min	ESAX representation of each segment									
				a=2	a=3	a=4	a=5	a=6	a=7	a=8	a=9	a=10	
Segment ₁	0.5772	0.5243	0.4612	bbb	ccc	ccc	ddd	eee	fee	fff	ggg	hhg	
Segment ₂	0	0	0	aaa	bbb	bbb	ccc	ccc	ddd	ddd	eee	eee	
Segment ₃	0.6621	0.6285	0.5916	bbb	ccc	ccc	ddd	eee	fff	fff	ggg	hhh	
Segment ₄	0.7976	0.7917	0.7729	bbb	ccc	ddd	ddd	eee	fff	ggg	hhh	hhh	
Segment ₅	0.7186	0.6930	0.6709	bbb	ccc	ddd	ddd	eee	fff	ggf	ggg	Hhh	
Segment ₆	0.4380	0.3285	0.1799	bbb	cbb	ccc	ddc	edd	eed	ffe	gff	ggf	
Segment ₇	0.6269	0.5864	0.5435	bbb	ccc	ccc	ddd	eee	ffe	fff	ggg	hhh	
Segment ₈	0.7941	0.7775	0.7580	bbb	ccc	ddd	ddd	eee	fff	ggg	hhh	hhh	
Segment ₉	0.7580	0.3003	0	bba	cbb	dcb	ddc	edc	fed	ged	gfe	hge	
Segment ₁₀	0.7644	0.3683	0	bba	cbb	dcb	ddc	edc	fed	gfd	hfe	hge	
Segment ₁₁	0.7373	0.3412	0	bba	cba	dcb	ddc	edc	fed	gfd	gfe	hge	
Segment ₁₂	0.7725	0.3919	0	bba	cbb	dcb	ddc	edc	fed	gfd	hfe	hge	
Segment ₁₃	0.7760	0.4402	0	bba	ccb	dcb	ddc	eec	fed	gfd	hge	hge	
Segment ₁₄	0.7832	0.5086	0	bba	ccb	dcb	ddc	eec	fec	gfd	hge	hge	
Segment ₁₅	0.7286	0.4240	0.1799	bbb	cbb	dcc	ddc	edd	fed	gfe	gff	hgf	
Segment ₁₆	0.7859	0.4877	0	bba	ccb	dcb	ddc	eec	fed	gfd	hge	hge	

Under this notation the ESAX word for the same segment can be written as $ESAX(s_{12}, 16, 2) = w_{12}^2 = \{bba\}$. Using the breakpoints we can derive the lower cardinality representation into the larger cardinality of the representation and vice versa for example: once we have w^{10} we can derive w^2 by simply with cardinality $a=2$ (Alphabet size= a, b) in which we have only two symbols (a and b) where any value ≤ 0 is represented by the symbol $\{a\}$ and any value > 0 is represented by the symbol $\{b\}$. All values represented by symbols $\geq f$ in w^{10} are always > 0 then it can be represented by the symbol $\{b\}$.

6. THE MATCHING PROCEDURE

The steps of our matching procedure are as follows:

- 1) Get the digital image and resize it to be in the normalized form so that each image has the same number of rows and columns.
- 2) Segmentation, if the image contains multi-objects we can get the connected components and label them by using the MATLAB function `bwlabel`.
For selecting a labeled object/shape and isolate it as a separate image, we set the gray level of all the points in the image to 255, except the points that constitute the required object, which will have the minimum gray level 0 (white) such that the extracted sub-image (object/shape) retains the same coordinates of that object/shape in the original image.
- 3) Align the major axis with x-axis.
- 4) Convert the object/shape into 1-D representation in the form of a time series, X , by using the

CAA/RAA technique.

- 5) Partition the pseudo time series or the vector $X=CAA/RAA$ into equal sized frames.
- 6) For each frame we compute minimum, mean and maximum values.

7. SIMILARITY MEASUREMENT

Our technique for measuring the similarity between two shapes is as follows:

- 1) Partition time series in both the query Q and the dataset T into equi-length segments.
- 2) Compute the max and min points, besides the mean value for data approximation in each segment.
- 3) For the task of retrieving approximately similar shapes within an image/shape database, we simply take each time series and represent it by symbols as described in section V. Find its distance to the query symbols.
- 4) The similarity between query image Q and database image I is determined as follows:

For each segment S_i represented by f symbols of each image I and image Q , the common distance measure used is the Euclidean distance which is defined as follows:

$$D = \sqrt{\sum_{i=1}^n (f_{I_i} - f_{Q_i})^2}, \text{ where } n \text{ is the number of segments.}$$

The following example shows the Euclidean Distance that is used for similarity measure:

$$\begin{array}{c} c \ a \ b \\ \uparrow \ \uparrow \ \uparrow \\ c \ b \ b \end{array} \text{ has Euclidean Distance: } D = (0^2 + 1^2 + 0^2)^{1/2} = 1$$

$$\begin{array}{c} c \ a \ b \\ \uparrow \ \uparrow \ \uparrow \\ b \ a \ a \end{array} \text{ has Euclidean Distance: } D = (1^2 + 0^2 + 1^2)^{1/2} = \sqrt{2}$$

- 5) Only the image I with smaller distance, under an arbitrary suitable threshold ϵ , will be chosen to match the query, i.e. if $D < \epsilon$, then I is similar to Q and retrieve it.

The retrieval is completely controlled by adjusting the threshold ϵ . We observed that, for very large threshold, lots of database images are retrieved, even when the images are dissimilar to each other, while for very small threshold, almost one image is retrieved which is the one used as a query itself.

8. EXPERIMENTAL RESULTS

Tables 4 and 5 show some results of applying SAX and ESAX algorithms with the applications of three different Cardinalities denoted by $a=4$, $a=10$ and $a=15$. From these tables, it becomes clear that for retrieving most plausible similar images from datasets, Extended SAX with large cardinality helps to detect important patterns and to improve detection accuracy.

In order to evaluate the performance of the proposed image matching procedure, we applied it on ageneral purposeWANGdatabase[19] containing 1,000 images of the Corel stock photo, in JPEG format of size. Using ageneral purposeWANGdatabase we do not need to align with x axis as in step 2 section 5. We resized all the images to be in size 192x128. We apply the SAX and ESAX

representation with the same subsequence length. We use precision and recall measures to measure the performance of the retrieval system which are defined as follows [7], [17]:

$$\text{Recall} = \frac{\text{Number of relevant shapes retrieved}}{\text{Total number of all relevant shapes}}$$





































$$\text{Precision} = \frac{\text{Number of relevant shapes retrieved}}{\text{Total number of all retrieved shapes}}$$

From this preliminary evaluation, we noticed that, for all the SAX and ESAX methods, as the threshold increases to certain limit, the relevant shapes increases and the precision decreases.

9. CONCLUSION

In this work, In order to retrieve similar images, we use the Euclidean distance to determine the similarity. We applied the proposed image matching procedure on more than one image dataset, representing different image types, to evaluate its performance. The experimental results showed that, given any type of query images, the proposed approach is effective in retrieving similar images and ESAX with large cardinality helps to detect important patterns and to improve detection accuracy.

Table 4: Results retrieval ,with different thresholds, of applying SAX and ESAX representations of time series data using the CAA method on 1000 different images dataset with the query image"600.jpg".

The query		Sample Retrievals											
		Cardinality	Thresh	0.0	1.0	1.7320			2.0000				
a=4	SAX												
	Thresh	0.0	2.1928	2.2803	2.7202	2.7928	2.8982	2.9325	3.2167	3.2310	3.3100	3.1022	
	ESAX												
a=10	Thresh	0.0	2.2360		2.6457	3.0000	3.3166		3.8729	4.0000		4.1231	
	SAX												



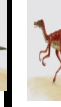
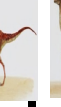







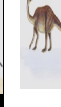



























	Thresh	0.0	3.6055	4.1231	4.2426	4.5825	4.6901	4.8765	4.9876	5.0990	5.2915	5.3643	5.5677	
	ESAX													
a=15	Thresh	0.0	2.4464	3.0000	3.4641	3.6055	3.8729	4.1231	4.2426	4.3588	4.4721	4.6904	4.8989	
	SAX													
	Thresh	0.0	3.9605	4.8885	5.1083	5.2074	5.3987	5.8594	5.8695	5.7537	6.3296	6.3460	6.4175	
	ESAX													

Table 5: Statistical results of applying SAX and ESAX representations of time series data using the CAA method on 1000 different images dataset with the query image "600.jpg" at a threshold $\epsilon \leq 7$.

ardinality C	Threshold ϵ		
	Method	$\epsilon \leq 7$	
a=4	SAX	Retrievals	350
		Relevant	25
		Recall	$25/100=0.25$
		Precision	$25/350=0.07$
	ESAX	Retrievals	350
		Relevant	64
		Recall	$54/100=0.54$
		Precision	$64/350=0.18$
a=10	SAX	Retrievals	66
		Relevant	19
		Recall	$19/100=0.19$
		Precision	$19/66=0.287$
	ESAX	Retrievals	21
		Relevant	16
		Recall	$16/100=0.16$
		Precision	$16/21=0.761$
a=15	SAX	Retrievals	64
		Relevant	17
		Recall	$17/100=0.17$
		Precision	$17/64=0.26$
	ESAX	Retrievals	21
		Relevant	17
		Recall	$17/100=0.17$
		Precision	$17/21=0.80$