doi:10.2498/cit.1001009

Affine Invariant Contour Descriptors Using Independent Component Analysis and Dyadic Wavelet Transform

Asad Ali, Syed Asif Mahmood Gilani and Nisar Ahmed Memon

Faculty of Computer Science and Engineering, GIK Institute of Engineering Sciences & Technology, NWFP, Pakistan

The paper presents a novel technique for affine invariant feature extraction with the view of object recognition based on parameterized contour. The proposed technique first normalizes an input image by removing the affine deformations using independent component analysis which also reduces the noise introduced during contour parameterization. Then four invariant functionals are constructed using the restored object contour, dyadic wavelet transform and conics in the context of wavelets. Experimental results are conducted using three different standard datasets to confirm the validity of the proposed technique. Beside this, the error rates obtained in terms of invariant stability are significantly lower when compared to other wavelet-based invariants. Also the proposed invariants exhibit higher feature disparity than the method of Fourier descriptors.

Keywords: affine invariants, independent component analysis, dyadic wavelet transform, conics, geometric transformations, pattern recognition

1. Introduction

One of the key tasks in robotic vision is to recognize objects when subjected to different viewpoint transformations. This can be achieved by constructing invariants to certain groups (Euclidean, affine, projective transformations) which hold potential for widespread applications for industrial part recognition [14], handwritten character recognition [15], identification of aircrafts [6], and shape analysis [16], to name a few. Viewpoint related changes of objects can broadly be represented by weak perspective transformation which occurs when the depth of an object along the line of sight is small compared to the viewing distance. This reduces the problem of perspective transformation to the affine transformation which is linear [18].

The affine group includes four basic forms of geometric deformations under weak perspective projection assumption, namely: translation, rotation, scaling and shearing. Finding a set of descriptors that can resist geometric attacks on the object contour can act as a good starting point for the more difficult projective group of transformations.

In this paper, we propose a new method of constructing invariants which is based on normalizing an affine-distorted and noise-corrupted object boundary using independent component analysis which makes it invariant to translation, scaling and shearing deformations, besides removing noise from the contour data points. Then, using the restored object contour we construct four invariants three of which use the approximation coefficients of the dyadic wavelet transform. It is important to mention here that the constructed invariants are independent of the contour scan order.

The paper is organized as follows. In Section 2 we briefly review the concepts of independent component analysis and dyadic wavelet transform. Next in Section 3 we review some of the previously published works, Section 4 describes the proposed method in detail and Section 5 provides experimental results and comparisons with previously published techniques.

2. Review

Before we move ahead, let us make a brief overview of the two primary components namely independent component analysis and dyadic wavelet transform that form the basis of the proposed technique.

2.1. Independent Component Analysis

Primarily developed to find a suitable representation of multivariate data, it performs blind source separation of a linear mixture of signals and has found numerous applications in short time. Assume that we observe a linear mixture Q of n independent components:

$$Q_j = A_{j1}S_1 + A_{j2}S_2 + \ldots + A_{jn}S_n \text{ for all } j \quad (1)$$

where A represents the mixing variable and S the source signals. Using vector notation it can be expressed as:

$$Q = AS. \qquad (2)$$

The model above is called the independent component analysis or ICA model [1][2] which is a generative model as it describes the process of mixing the component signals S_i. All that is observed is Q and A, S must be estimated from it. In order to estimate A, the component S_i must be statistically independent and have a non-gaussian distribution. After estimating the mixing variable A we can compute its inverse say W and obtain the independent components as:

$$S = WQ. \qquad (3)$$

We opted for ICA as a possible solution space because an affine deformation of the object contour results in the linear mixing of the data points on the coordinate axis besides being coupled with random noise during contour parameterization.

2.2. Dyadic Wavelet Transform

In pattern recognition, it is important to construct signal representations that are invariant to translation. Whenever a pattern undergoes a shift, the descriptors may also undergo a shift, but should not change its numeric signature. Continuous wavelet transform and windowed Fourier transform provide translation invariant representations, but uniformly sampling the translation parameter destroys this translation invariance. Whereas the dyadic wavelet transforms maintain translation invariance by sampling only the scale parameter along a dyadic sequence $\{2^j\}_{j\in Z}$ of the continuous wavelet transform [19]. Thus the scale parameter is discretized, but not the translation parameter. Hence the resulting input signal may undergo a shift, but the sampling mechanism of the dyadic wavelet transform preserves the numeric values from being modified due to shift in the input signal.

3. Related Work

Keeping in view the importance of constructing invariants and their widespread applications, research has been conducted by many which can broadly be classified into two groups, namely: Region-based and Contour-based invariant descriptors. In the context below we review some of the contour-based techniques that are most related to the present work.

Several parameterizations of the object boundary that are linear under an affine transformation have been proposed. The affine arc length τ proposed in [8] is defined as follows:

$$\tau = \int_a^b \sqrt[3]{x(t)'y(t)'' - x(t)''y(t)'}dt \qquad (4)$$

where $x(t)'$, $y(t)'$ are the first and $x(t)''$, $y(t)''$ the second order derivatives with respect to the parameterization order t. As the above computation requires second order derivates, it becomes susceptible to noise introduced because of incorrect segmentation of the object.

To solve the above problem Arbter et al. [9] introduced the invariant Fourier descriptors using the enclosed area parameter defined as:

$$\sigma = \frac{1}{2}\int_a^b |x(t)y(t)' - y(t)x(t)'|dt \qquad (5)$$

The above formulation was derived using the property that the area occupied by an object changes linearly under an affine transformation. The only drawback is that it is not invariant to translation and requires the starting and ending points to be connected. Arbter [9] also found that using sign in the enclosed area parameter (5) makes it much less sensitive to noise instead of the absolute values. Besides this, compared to the wavelet-based descriptors, this technique has a higher misclassification rate.

Zhao et al. [10] introduced affine curve moment invariants based on affine arc length (4) defined as:

$$v_{pq} = \int_C [x(t) - \tilde{x}]^p [y(t) - \tilde{y}]^q \{[x(t) - \tilde{x}]y(t)'[y(t) - \tilde{y}]x(t)'\}dt \tag{6}$$

where $\tilde{x}$ and $\tilde{y}$ are the centroid of the contour computed using (4) after removing the cubic root in the framework of moments. They derived a total of three invariants using equation (6) and have shown them to be invariant to the affine group of transformations. The drawback of the above framework is that the invariants are sensitive to noise and local variations of shape because the computation of invariants is based on moments and derivates of first order.

More recently, Manay et al. [7] introduced the Euclidean integral invariants to counter the effect of noise based on the concept of differential invariants. They have derived two invariants, namely; distance integral invariant and area integral invariant. The major drawback of their work is that the distance integral invariant is a global descriptor and a local change of shape i.e. missing parts of shape, affects the invariant values for the entire shape, whereas the area integral invariant only counters for the Euclidean group of transformations.

Tieng et al. [4] proposed the use of dyadic wavelet transform for constructing invariants using the approximation and detail coefficients. They formulated a framework based on enclosed area parameter for constructing invariants in the wavelet domain. Later Khalil et al. [5][6] extended their work and derived invariants using the detail coefficients and wavelet-based conic equation.

More recently, Ibrahim et al. [3] derived invariants using the approximation coefficients based on the framework proposed in [4] and showed that approximation-based invariants outperform detail-based invariants in terms of error rates. In this paper we extend our work initially proposed in [21] and add two more invariants to the descriptor list besides elaborating in detail the experimental results. In the process, we make use of the frameworks proposed in [4] and [6] while constructing invariants in the next sections and improve upon the wavelet-based methods by reducing error rates.

4. Proposed Technique

We propose a three-step process for the construction of contour-based invariant descriptors of the objects. The first step acts as foundation for second and third steps (divided into two parts: construction of invariant I$_1$, I$_2$ and I$_3$, I$_4$) in which ICA is applied and then invariants are constructed. Next we provide the detailed description of each step.

4.1. Boundary Parameterization and Re-sampling

In the first step, object contour is extracted and parameterized. Let us define this parametric curve as $[x(t), y(t)]$ with parameter t on a plane. Next the parameterized boundary is resampled to a total of L data points. Thus a point on the resampled curve under an affine transformation can be expressed as:

$$\begin{aligned} \tilde{x}(t) &= a_0 + a_1 x(t) + a_2 y(t) \\ \tilde{y}(t) &= b_0 + b_1 x(t) + b_2 y(t) \end{aligned} \tag{7}$$

The above equations can be written in matrix form as:

$$\begin{bmatrix} \tilde{x}(t') \\ \tilde{y}(t') \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} + \begin{bmatrix} a_0 \\ b_0 \end{bmatrix}$$

$$\begin{bmatrix} \tilde{x}(t') \\ \tilde{y}(t') \end{bmatrix} = P \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} + B \tag{8}$$

$$Y'(t') = PY(t) + B$$

where t and t' are different because of the difference in contour scan order and sampling of the two contours, Y' is obtained as a result of affine transformation of Y, P is the affine transformation matrix and B is the translation vector which can be removed ($B = 0$) by using the centroid contour coordinates.

The source code written for contour parameterization can be obtained from the following weblink[1].

[1] http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=13109&objectType=file

4.2. Theoretical Formulation and Application of ICA

We know that $Y(t)$ and $Y'(t')$ are the linear combination of the same source S with a different mixing matrix A and A' referring to equation (2). Then we can write:

$$\begin{aligned} Y(t) &= AS(t) \\ Y'(t') &= A'S(t) \end{aligned} \tag{9}$$

where A' is the linear combination of P and random noise $\mathbf{N}$. In (9) the mixing matrix A' is different because of the difference in affine transformation parameters and the random noise introduced during contour parameterization.

Next we estimate the mixing variable A' by finding a matrix W of weights using the Fast ICA algorithm from [1]. Then W will be used to find the original source S as per equation (3). The two-step process for computing ICA is as follows:

Step 1: Whiten the Centered Data

Whitening is performed on $Y'(t')$ in order to reduce the number of parameters that need to be estimated. Its utility resides in the fact that the new mixing matrix $\tilde{A}'$ that will be estimated is orthogonal such that it satisfies:

$$\tilde{A}'\tilde{A}'^{\mathrm{T}} = I \tag{10}$$

So, the data Y' becomes uncorrelated after this step. Whitening is stronger than uncorrelatedness as whiteness of a zero mean vector means that its components are uncorrelated and their variance equals unity. In other words, the covariance and correlation matrix of $\tilde{A}'$ equals the identity matrix as per equation (10). Whitening is then performed by computing the Eigen value decomposition of covariance matrix as:

$$\begin{aligned} Y'Y'^{\mathrm{T}} &= EDE^{\mathrm{T}} \\ \tilde{Y}' &= ED^{-1/2}E^{\mathrm{T}}Y' \end{aligned} \tag{11}$$

where E is the orthogonal matrix of eigenvectors of $\{Y'Y'^{\mathrm{T}}\}$ and D is the diagonal matrix of eigen values.

Step 2: Apply ICA on the Whitened Object Contour

Here we apply the independent component analysis on the whitened contour $\tilde{Y}' = [x'(t')\, y'(t')]$. The steps involved in the algorithm are detailed below:

a. Initialize a random matrix of weights W.

b. Compute the intermediate matrix as:

$$W^{+} = E\{\tilde{Y}'g(W^{\mathrm{T}}\tilde{Y}')\} - E\{\tilde{Y}'g'(W^{\mathrm{T}}\tilde{Y}')\}W \tag{12}$$

where g is a non quadratic function and $E\{.\}$ represents the maxima of the approximation of negentropy. For more details refer to [1].

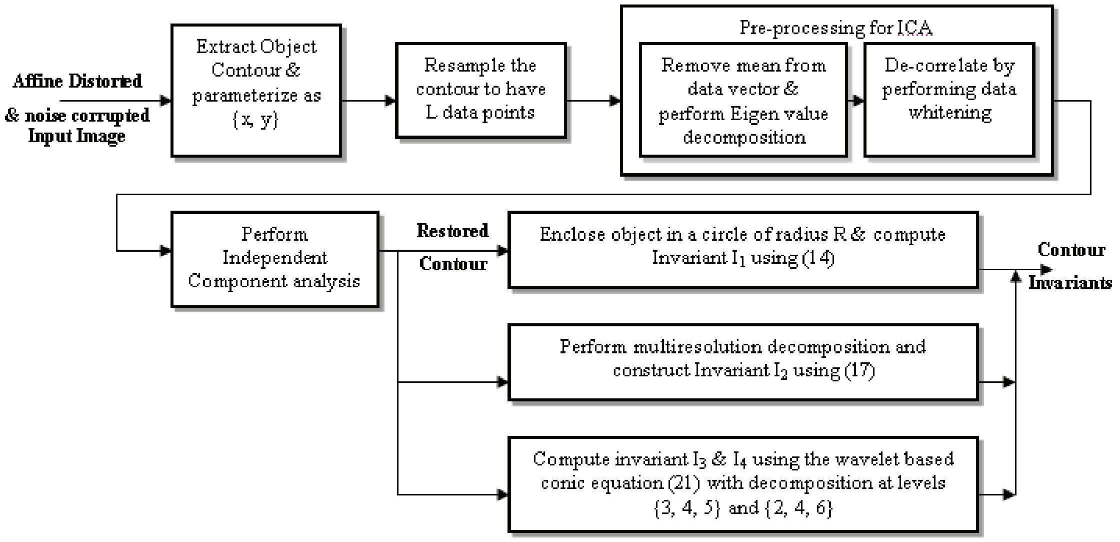

Affine Distorted
& noise corrupted
Input Image
Extract Object Contour & parameterize as {x, y}
Resample the contour to have L data points
Pre-processing for ICA
Remove mean from data vector & perform Eigen value decomposition
De-correlate by performing data whitening
Perform Independent Component analysis
Restored Contour
Enclose object in a circle of radius R & compute Invariant I1 using (14)
Perform multiresolution decomposition and construct Invariant I2 using (17)
Compute invariant I3 & I4 using the wavelet based conic equation (21) with decomposition at levels {3, 4, 5} and {2, 4, 6}
Contour Invariants

Figure 1. The complete system diagram for the construction of contour-based invariant descriptors.

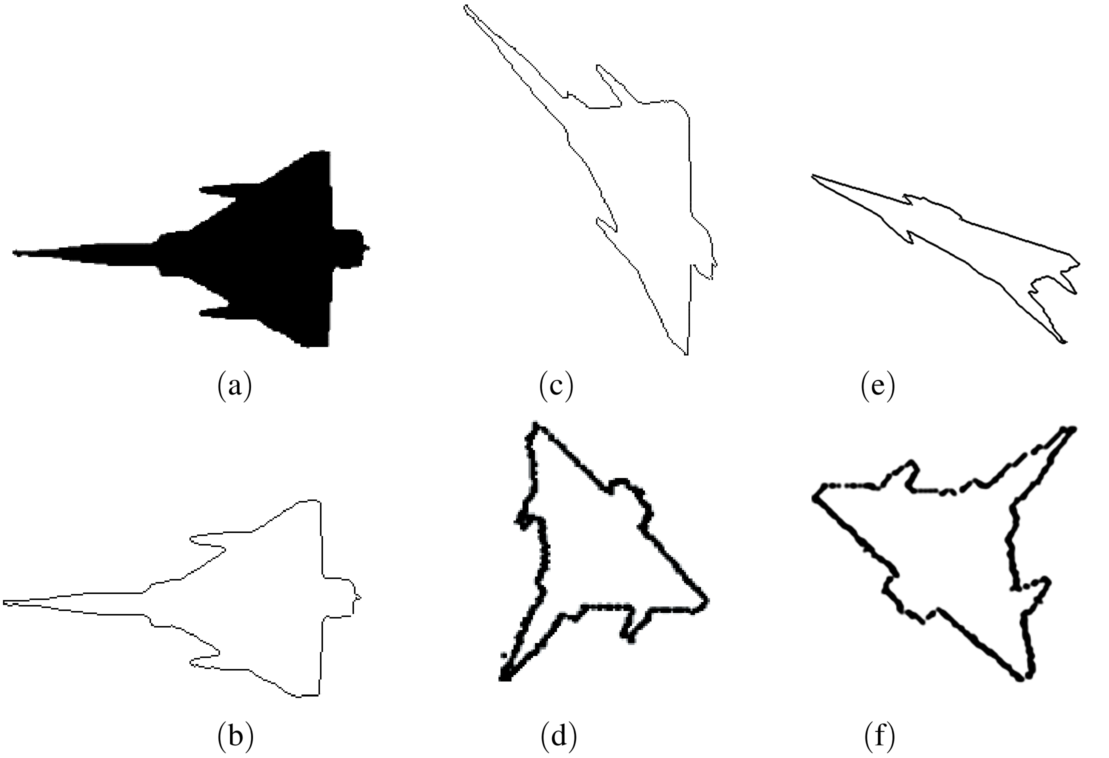


Figure 2. (a) Original image. (b) Parameterized boundary. (c) and (e) are affine transformed version of (b). (d) and (f) are the restored (normalized) counterparts obtained after performing whitening.

c. Let $W = W^{+}/||W^{+}||$

d. If not converged, then go back to (b).

It is important to note that convergence means that the previous and current values of W have the same sign and the difference is below a certain permissible value.

By using the above procedure we have been able to find a matrix W' of weights that satisfies:

$$\begin{aligned} W'Y'(t') &= W'AS(t') \approx S(t') \\ A' &= W'^{-1} \end{aligned} \tag{13}$$

So we now use the inverse of the matrix W' to find S as per equation (3). The obtained source $S(t')$ will have the same statistical characteristics as the original source $S(t)$, but will only differ from it because of the random contour parameterization order.

Figure 1 shows the complete system diagram and elaborates the above mentioned operations in a sequential and precise manner where, as Figure 2 and Figure 3 demonstrate, the output obtained after applying the above mentioned steps.

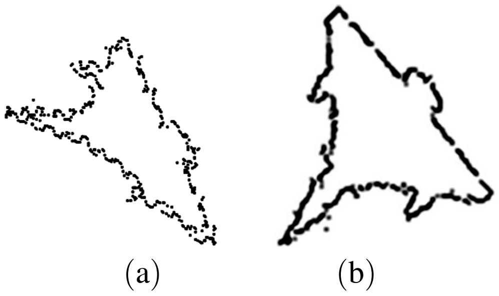


Figure 3. (a) An affine deformed and noise-corrupted object contour. (b) Noise-reduced and affine normalized image obtained after applying ICA.

Although using the above procedure we have been able to recover the contour of the object, the obtained independent components may have been inverted along either the parameterized x-axes or y-axes. As a result there are four possible cases $[x, y]$, $[x^r, y]$, $[x, y^r]$ and $[x^r, y^r]$ where x^r, y^r represent values in reverse order. However, we can consider only one of the two cases $[x, y]$ and $[x^r, y^r]$ for invariant construction as the effect of inversion along both axes can be removed by using normalized cross correlation. So we are left with three cases and we construct invariants I_1, I_2, I_3 and I_4 proposed in the next sub-section for each of the cases and use them while performing cross correlation.

4.3. Affine Invariant Functions I_1 & I_2

As a consequence of previous operations we have been able to remove translation, scaling and shearing distortions from the object contour, besides considerably reducing the effect of noise which is introduced during the parameterization process because of incorrect segmentation. The only distortion we are left with is rotation. So in this third and final step we construct two invariants using the restored object contour.

Stepwise process used for the construction of invariant I_1 is described below:

a. Enclose the full object contour in a circle C = [C_x C_y] of radius R as shown in Figure 4(c).

b. Compute the Manhattan distance of a point lying on the recovered object contour S(t′) = [x(t′) y(t′)] with all the points of C as:

$$D = |x_{t'} - C_x| + |y_{t'} - C_y| \qquad (14)$$

c. Select the minimum value of D as an invariant and add it to $I_1(t)$.

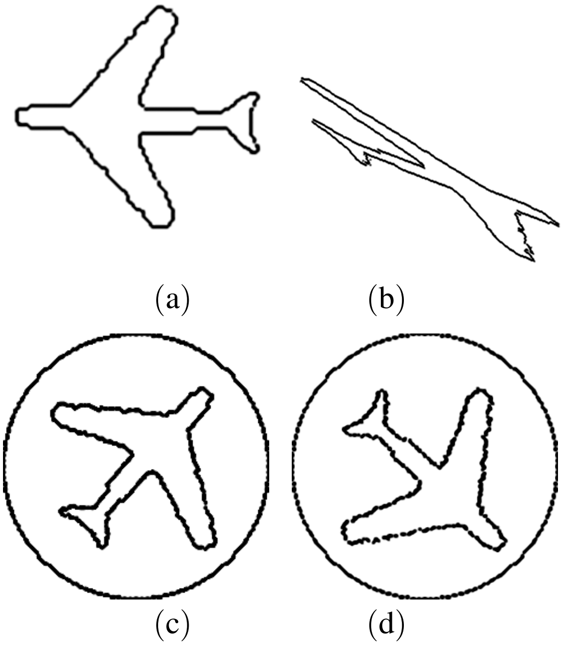


Figure 4. (a) Original parameterized boundary. (b) Affine transformed object. (c), (d) Enclosed objects after restoration for constructing invariant I_1.

d. Repeat (b) and (c) for every point lying on the object contour.

As a result of the above operation we have been able to convert rotational distortion of an object into translational misalignment which can be removed by performing normalized cross correlation of the original and transformed object contour parameterized in an unknown order. Figure 5[(a), (b)] shows the plots of Invariant I_1 for the same object and its deformed version. Normalized cross correlation value obtained for invariant I_1 in Figure 5 is 0.9627. Besides this, the invariant can also resist small deformations such as missing parts of shape.

In order to increase the discriminative capability of the invariants we make use of the wavelet transform to construct invariant I_2. Wavelet transform is a linear transform and if it is applied to the affine-distorted shape then it also gets affected by the same distortion. But the object in our case is only rotationally deformed and all other geometric deformations are removed. Then we can write:

$$T = \begin{bmatrix} WT_ix(t') & WT_jx(t') \\ WT_iy(t') & WT_jy(t') \end{bmatrix}$$

$$T = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} T \qquad (15)$$

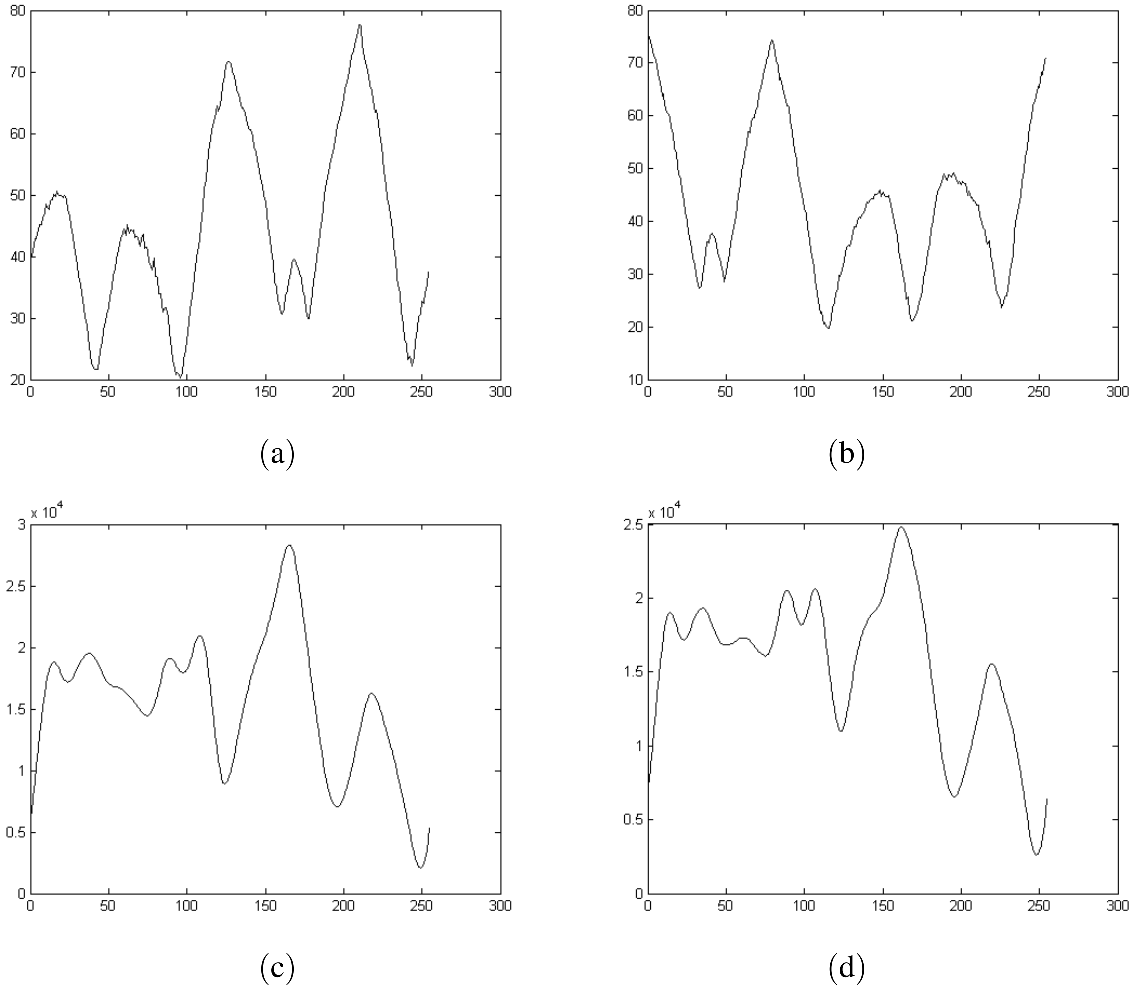
An affine invariant function is computed by taking the determinant of (15) as:

$$WT_ix(t')WT_jy(t') - WT_iy(t')WT_jx(t') = \det(V)(WT_ix(t')WT_jy(t') - WT_iy(t')WT_jx(t')) \qquad (16)$$

where V is the rotational transformation matrix which only effects the contour parameterization order, i and j represent coefficients at two different levels of the wavelet transform. In (16), only if the approximation coefficients of the wavelet transform are used, it can be written as:

$$I_2 = A_ix(t')A_jy(t') - A_iy(t')A_jx(t') \qquad (17)$$

To construct the above invariant we make use of the "A Trous algorithm" proposed by Mallat [19]. Figure 5[(c), (d)] shows the plot of I_2 for the image in 4(a) and 4(b).



(a) (b) (c) (d)

Figure 5. (a), (c) Invariant I_1 and I_2 for the object shown in Figure 4(a).
(b), (d) Invariant I_1 and I_2 for the corresponding affine deformed object shown in Figure 4(b).

4.4. Affine Invariant Functions I_3 & I_4

Here we construct two invariants by using conics in context of the dyadic wavelet transform for the restored object contour. Conics have been used previously in computer vision to derive geometric invariant functions. Conics are curves defined in terms of projective invariant property and have a central role in projective geometry. A geometric interpretation of these invariants is given in [18].

For a point (x, y) from the restored object contour the conic can be expressed as the quadratic form [20]:

$$[x \quad y]\, G \begin{bmatrix} x \\ y \end{bmatrix} = h,$$

$$\text{where } G = \begin{bmatrix} G_{11} & G_{12} \\ G_{12} & G_{22} \end{bmatrix} \tag{18}$$

where h is a constant and G is a symmetric matrix.

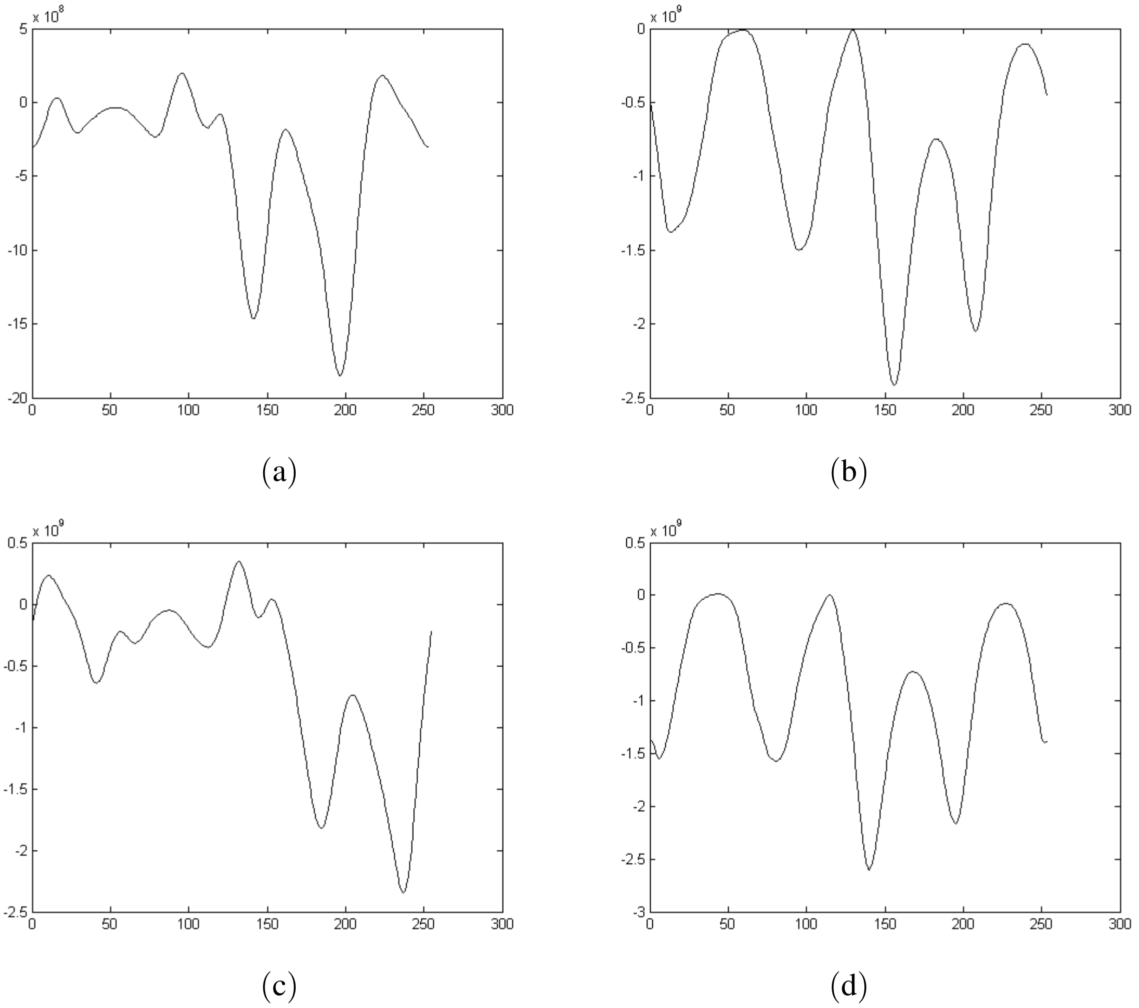
A wavelet-based conic equation [6] can be obtained from (18) using three dyadic levels $W_i x(t)$ and $W_i y(t)$ where W represents the wavelet transform and $i \in \{a, b, c\}$.

$$[\, W_i x(t) \qquad W_i y(t)\,]\, \zeta(t) \begin{bmatrix} W_i x(t) \\ W_i y(t) \end{bmatrix} = h$$

$$\text{where } \zeta = \begin{bmatrix} \eta_{11} & \eta_{12} \\ \eta_{12} & \eta_{22} \end{bmatrix} \tag{19}$$

An affine invariant function can then be defined as:

$$\eta_{a,b,c}(t) = \eta_{11}(t)\eta_{22}(t) - \eta_{12}^2(t) \tag{20}$$



(a) (b)

(c) (d)

Figure 6. (a), (b) Invariant I_3 and I_4 for the object shown in Figure 4(a).
(c), (d) Invariant I_3 and I_4 for the corresponding affine deformed object shown in Figure 4(b).

The above function has been proven by [5][6] to be equivalent to:

$$\begin{aligned}\eta_{a,b,c} = & -f^4_{c,b}(t) - f^4_{a,c}(t) - f^4_{b,a}(t) \\ & + 2f^2_{a,c}(t)f^2_{c,b}(t) + 2f^2_{c,b}(t)f^2_{b,a}(t) \\ & + 2f^2_{b,a}(t)f^2_{a,c}(t)\end{aligned} \tag{21}$$

where

$$f_{p,q} = A_p x(t) A_q y(t) - A_q x(t) A_p y(t) \tag{22}$$

The function in (21) is an invariant of weight four. Affine invariant functions I_3 and I_4 can then be constructed as:

$$\begin{aligned}I_{[3,4]} = & -f^4_{c,b}(t) - f^4_{a,c}(t) - f^4_{b,a}(t) \\ & + 2f^2_{a,c}(t)f^2_{c,b}(t) + 2f^2_{c,b}(t)f^2_{b,a}(t) \\ & + 2f^2_{b,a}(t)f^2_{a,c}(t).\end{aligned} \tag{23}$$

It is clear from here that (23) is itself a function of (22) which is affine invariant. We make use of the approximation coefficients of the dyadic wavelet transform while constructing invariants I_3 and I_4 using (23). The dyadic wavelet transform is implemented using the "A Trous algorithm" proposed by Mallat [19]. Figure 6 shows the plot of invariants I_3 and I_4.

5. Experimental Results

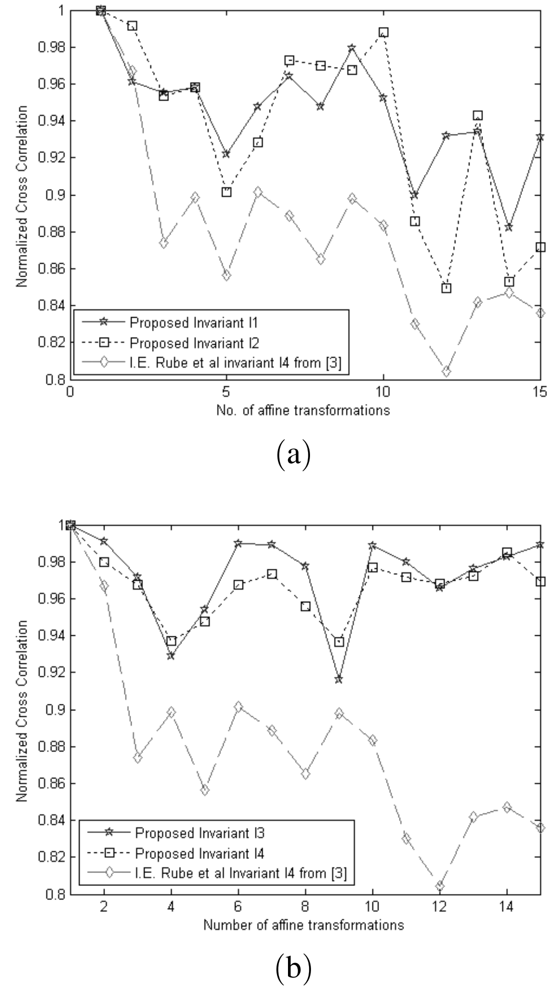
The proposed technique was tested on a 2.4 GHz Pentium IV machine with Windows XP and Matlab as the development tool. The datasets used in the experiments include the MPEG-7 Shape-B datasets, 10 aircraft images from [6] and English alphabets dataset. All the parameterized contours are resampled to have the same length L of 256 data points. In the construction of the invariant I_1 the value of R used is 80, qubic spline filters for the construction of invariant I_2 have been used with decomposition up to level four, where, as in the construction of invariant I_3 and I_4, the approximations coefficients at level $\{3,4,5\}$ and $\{2,4,6\}$ are used. As the invariants constructed in section (4.4) are based on the same mathematical concept hence by varying the wavelet decomposition levels only a large number of invariants can be constructed as has been the case for invariants I_3 and I_4. Further qubic spline filters are used for wavelet decomposition.

The decomposition levels were selected to be slightly coarser. The purpose was to prevent the invariants from small noise perturbations left on the output of ICA which can also mimic itself in the output of wavelet transform if a finer decomposition level is selected. Whereas selecting very coarse decomposition levels can reduce the discrimination capability of constructed descriptors between objects resembling outer contours. Hence a tradeoff between sensitivity to noise distortions and discrimination capability resulted in the selection of above decomposition levels.

Besides this, we use normalized cross correlation for comparing two sequences A_k and B_k which is defined as:

$$R_{AB} = \frac{\sum_{l}\sum_{k} A_k B_{k-l}}{\sqrt{\sum_{k} A_k^2 \sum_{k} B_k^2}} \tag{24}$$

This section is divided into three parts: first we demonstrate the stability of the four invariants against five different affine transformations then we provide a comparative analysis of the four invariants with the method in [3] and lastly we demonstrate the feature discrimination capability of the four invariants when compared to the method of Fourier descriptors.



(a)

(b)

Figure 7. The comparison of invariant I_1, I_2, I_3 and I_4 with the method in [3]. The results are averaged over the MPEG-7 shape-B dataset.

Transformation	Object 1 [figure 2(b)]				Object 2 [figure 4(a)]			
	I_1	I_2	I_3	I_4	I_1	I_2	I_3	I_4
Original Image	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
R(70), S(2,1)	0.9447	0.9562	0.9693	0.9571	0.9712	0.9718	0.9403	0.9335
R(135), S(2,3), T	0.9850	0.9634	0.9718	0.9709	0.9854	0.9936	0.9785	0.9596
R(45),Sh(2.05,1.0),T	0.9124	0.8550	0.9369	0.9202	0.9632	0.8949	0.9035	0.9267
R(165), S(3,3), Sh(1,2), T	0.9309	0.9070	0.9845	0.9818	0.9058	0.9290	0.9148	0.9423
R(230), S(4,1), Sh(3,3), T	0.9165	0.8561	0.9376	0.9679	0.9301	0.8641	0.9217	0.9466

Table 1. The normalized cross correlation values of the invariants after applying different affine transformations.

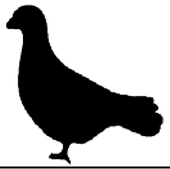
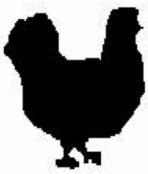
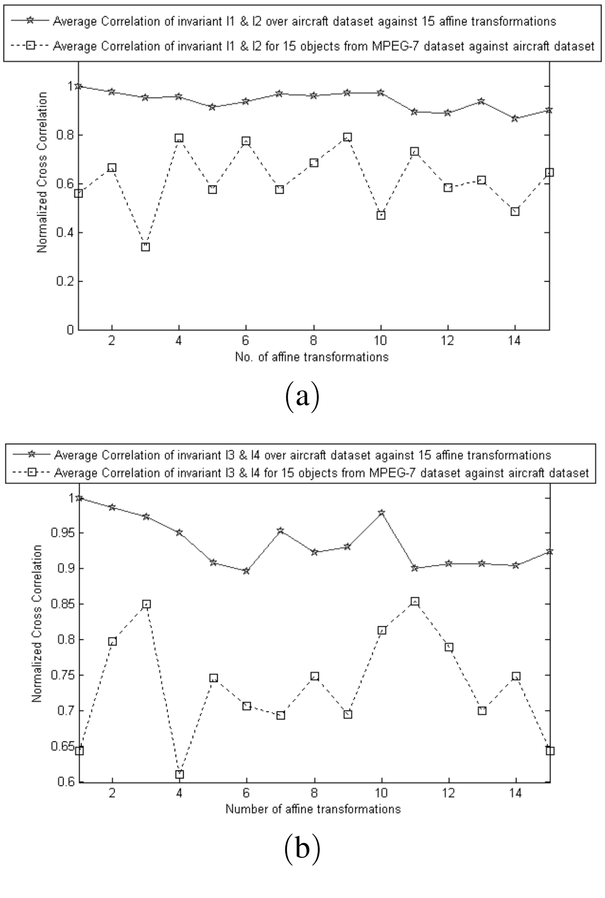
S.No	Image Name	Transformation	I_1	I_2	I_3	I_4
1.	Butterfly 	Original Image	1.00	1.00	1.00	1.00
		R(70), S(2,1)	0.9432	0.9522	0.9916	0.9828
		R(135), S(2,3), T	0.9534	0.9961	0.9897	0.9714
		R(45),Sh(2.05,1.0),T	0.9460	0.9561	0.9834	0.9627
		R(165), S(3,3), Sh(1,2), T	0.9424	0.9508	0.9864	0.9771
		R(230), S(4,1), Sh(3,3), T	0.9272	0.9658	0.9865	0.9648
2.	Bird 	Original Image	1.00	1.00	1.00	1.00
		R(70), S(2,1)	0.9279	0.9757	0.9285	0.9473
		R(135), S(2,3), T	0.9484	0.9868	0.9692	0.9012
		R(45),Sh(2.05,1.0),T	0.8747	0.9018	0.9053	0.8929
		R(165), S(3,3), Sh(1,2), T	0.9273	0.9761	0.9033	0.9304
		R(230), S(4,1), Sh(3,3), T	0.8826	0.8996	0.9148	0.9180
3.	Device 	Original Image	1.00	1.00	1.00	1.00
		R(70), S(2,1)	0.8847	0.9203	0.9185	0.9182
		R(135), S(2,3), T	0.9619	0.9477	0.9678	0.8944
		R(45),Sh(2.05,1.0),T	0.8926	0.8784	0.8894	0.9382
		R(165), S(3,3), Sh(1,2), T	0.9200	0.9310	0.9508	0.9631
		R(230), S(4,1), Sh(3,3), T	0.9081	0.9160	0.9085	0.9090
4.	Chopper 	Original Image	1.00	1.00	1.00	1.00
		R(70), S(2,1)	0.9211	0.8817	0.9504	0.9310
		R(135), S(2,3), T	0.9447	0.9231	0.9675	0.9333
		R(45),Sh(2.05,1.0),T	0.8830	0.8710	0.8890	0.8917
		R(165), S(3,3), Sh(1,2), T	0.9217	0.8958	0.9631	0.9516
		R(230), S(4,1), Sh(3,3), T	0.8595	0.9129	0.9465	0.8864
5.	Bat 	Original Image	1.00	1.00	1.00	1.00
		R(70), S(2,1)	0.9031	0.9631	0.8711	0.8853
		R(135), S(2,3), T	0.9276	0.9944	0.9241	0.9231
		R(45),Sh(2.05,1.0),T	0.9031	0.9359	0.9257	0.9754
		R(165), S(3,3), Sh(1,2), T	0.9257	0.9768	0.9142	0.9427
		R(230), S(4,1), Sh(3,3), T	0.8844	0.9571	0.9478	0.9425
6.	Camel 	Original Image	1.00	1.00	1.00	1.00
		R(70), S(2,1)	0.8900	0.9060	0.9341	0.9487
		R(135), S(2,3), T	0.9725	0.8975	0.9479	0.9066
		R(45),Sh(2.05,1.0),T	0.9138	0.9580	0.9609	0.9750
		R(165), S(3,3), Sh(1,2), T	0.9382	0.9143	0.9721	0.9426
		R(230), S(4,1), Sh(3,3), T	0.9233	0.9118	0.8712	0.8984
7.	Chicken 	Original Image	1.00	1.00	1.00	1.00
		R(70), S(2,1)	0.9375	0.9683	0.9394	0.9653
		R(135), S(2,3), T	0.9364	0.9191	0.9581	0.9651
		R(45),Sh(2.05,1.0),T	0.8959	0.8903	0.9368	0.9376
		R(165), S(3,3), Sh(1,2), T	0.9197	0.9780	0.9429	0.9596
		R(230), S(4,1), Sh(3,3), T	0.8834	0.8904	0.9465	0.9542
8.	Apple 	Original Image	1.00	1.00	1.00	1.00
		R(70), S(2,1)	0.9180	0.9985	0.9773	0.9713
		R(135), S(2,3), T	0.9513	0.9642	0.9805	0.9720
		R(45),Sh(2.05,1.0),T	0.9357	0.9405	0.9451	0.9375
		R(165), S(3,3), Sh(1,2), T	0.9813	0.9763	0.9774	0.9527
		R(230), S(4,1), Sh(3,3), T	0.9353	0.9088	0.8743	0.9130

Table 2. The normalized cross correlation values of the invariants after applying different affine transformations on a subset of MPEG-7 shape-B dataset.

Table 1 provides comparison of the invariants I_1, I_2, I_3 and I_4 in terms of the normalized cross correlation values against different affine transformations for the objects in Figure 2(b) and Figure 4(a) from the aircraft dataset. In the table, the following notation is used: Rotation (R) in degrees, Scaling (S), Shear (Sh) along x and y axis and Translation (T). The figures in brackets represent the parameters of the transformation.

Similarly, Table 2 provides the comparison of invariants I_1, I_2, I_3 and I_4 in terms of the normalized cross correlation values against different affine transformations for a selected set of objects from the MPEG-7 Shape-B dataset.

To further elaborate and demonstrate invariant stability, Figure 7 compares the proposed invariants I_1, I_2, I_3 and I_4 with [3] over a set of 15 affine transformations shown in Table 3. The results are averaged over the MPEG-7 shape-B dataset. Obtained results show a significant increase in performance as a function of increased correlation between the original and affine transformed images for the proposed invariants.



(a)

(b)

Figure 8. The discrimination capability of invariant I_1, I_2, I_3 and I_4 using the aircraft and MPEG7 dataset.

S.No	Rotation$^\circ$	Scaling X	Scaling Y	Shear X	Shear Y
1.	35	3	3	0	0
2.	150	3	4	0	0
3.	170	3	2	1	0
4.	140	1	4	0	0
5.	30	1	2	0	0
6.	70	3	2	0	1
7.	150	1	3	2	0
8.	235	1	4	1	0
9.	46	4	2	0	0
10.	45	1	1	1	3
11.	55	1	1	3	1
12.	65	1	1	2.05	2
13.	110	1	1	1.5	1.7
14.	25	3	3	1.5	1.7
15.	120	2	1	0	1

Table 3. The set of fifteen affine transformations used for performing various experiments.

Finally we demonstrate the feature discrimination capability of the proposed invariants using Figure 8 and compare it with that of the Fourier Descriptors in Figure 9. Figure 8[(a), (b)] plots the result of correlation of the proposed invariants for the aircraft dataset and its fifteen affine transformed versions and correlation of fifteen objects and there affine transformed version from the MPEG-7 shape-B dataset with the aircraft dataset.

The results have been averaged for I_1, I_2 (Figure 8(a)) and I_3, I_4 (Figure 8(b)). For the invariants that can exhibit good disparity between

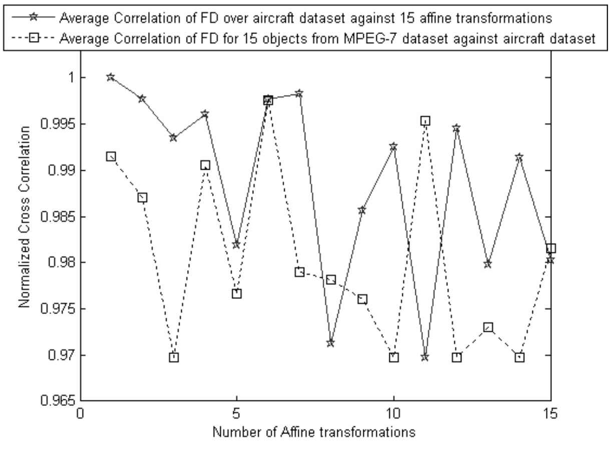


Figure 9. The discrimination capability of Fourier Descriptors using the aircraft and MPEG7 dataset.

shapes the two correlation plots should not overlap which has been the case for the proposed invariants I_1, I_2 and I_3, I_4 in Figure 8. Figure 9 plots the above mentioned correlations using the method of Fourier Descriptors where the two correlation plots overlap significantly.

Although the resampling of object contour results in loss of may be precious data points and can have a negative effect on cross correlation values but at the same time it is necessary so that the signals to be compared don't become the subset of each other and correspondence can be established fairly. Further processing time is reduced substantially against arbitrary length signals.

It is important to mention here that a preprocessing step such as a smoothing operation applied on the object contour after restoration can significantly increase the correlation values, which at present has not been used to preserve the shape discrimination power of the two invariants. Obtained results show significant reduction in error, thus validating the proposed approach.

6. Conclusion

In this paper we have presented a new technique for invariant construction using the independent component analysis and dyadic wavelet transform. Experimental results validate the use of an affine normalization technique as a preprocessor to the computation of invariant functionals. Besides this, the use of dyadic wavelet transform after affine normalization added the much needed discriminative power to the proposed set of invariants. Presently, our work is in progress to extend the framework to handle the projective group of transformations and estimation of the affine parameters. In future, we intend to build an intelligent classifier for performing object recognition over a large dataset based on the proposed invariants.

Acknowledgment

The authors would like to thank National Engineering and Scientific Commission (NESCOM) for their financial support, GIK Institute of Engineering Sciences & Technology for facilitating this research and Temple University, USA for providing the MPEG-7 Shape-B dataset.

References

[1] A. HYVARINEN, Fast and robust fixed point algorithms for independent component analysis. *IEEE Transactions on Neural Network*, **10**(3), May 1999.

[2] A. HYVARINEN, E. OJA, A fast fixed point algorithm for independent component analysis. *Neural Computations*, **9**, October 1997.

[3] I. E. RUBE, M. AHMED, M. KAMEL, Wavelet approximation-based affine invariant shape representation functions. *IEEE Transactions on pattern analysis and machine intelligence*, **28**(2), February 2006.

[4] Q. TIENG, W. BOLES, An application of wavelet-based affine invariant representation. *Pattern recognition Letters* **16**(12), December 1995.

[5] M. I. KHALIL, M. BAYOUMI, Affine invariants for object recognition using the wavelet transform. *Pattern recognition letters*, **23**(1–3), January 2002.

[6] M. KHALIL, M. BAYOUMI, A dyadic wavelet affine invariant function for 2D shape recognition. *IEEE Transactions on pattern analysis and machine intelligence*, **23**(10), October 2001.

[7] S. MANAY, D. CREMERS, B. HONG, A. YEZZI, S. SOATTO, Integral Invariants for Shape Matching. *IEEE Transactions on pattern analysis and machine intelligence*, **28**(10), October 2006.

[8] H. W. GUGGENHEIMER, *Differential Geometry*, McGraw-Hill, New York, 1963.

[9] K. ARBTER, E. SYNDER, H. BURKHARDT, G. HIRZINGER, Application of affine invariant Fourier descriptors to the recognition of 3D objects. *IEEE Transactions on pattern analysis and machine intelligence*, **12**(7), July 1990.

[10] D. ZHAO, J. CHEN, Affine curve moment invariants for shape recognition. *Pattern recognition,* **30**(6), June 1997.

[11] J. FLUSSER, T. SUK, Pattern recognition by affine moment invariants. *Pattern recognition*, **26**(1), January 1993.

[12] M. PETROU, A. KADYROV, Affine invariant features from the trace transform. *IEEE transactions on pattern analysis and machine intelligence*, **26**(1), January 2004.

[13] E. RAHTU, M. SALO, J. HEIKKILA, Affine invariant pattern recognition using multiscale autoconvolution. *IEEE Transactions on pattern analysis and machine intelligence*, **27**(6), June 2005.

[14] Y. LAMDAN, J. T. SCHWARTZ, Affine Invariant Model-based Object Recognition. *IEEE Transactions on robotics and automation*, **6**(5), October 1990.

[15] T. WAKAHARA, K. ADAKA, Adaptive Normalization of handwritten characters using global-local affine transformations. *IEEE Transactions on pattern analysis and machine intelligence*, **20**(12), December 1998.

[16] I. E. RUBE, M. AHMED, M. KAMEL, Coarse to fine multiscale affine invariant shape matching and classification. *Proc of* 17*th* *International Conference on Pattern recognition*, August 2004.

[17] Z. HAUANG, F. S. COHEN, Affine invariant B-spline moments for curve matching. *IEEE Transactions on image processing*, **5**(10), October 1996.

[18] J. MUNDY, A. ZISSERMAN, *Geometric invariance in computer vision*, MIT Press, Cambridge, MA.

[19] S. MALLAT, *A Wavelet Tour of Signal Processing*, 2*nd* Edition, Academic Press, 1999.

[20] I. WEISS, Geometric invariants and object recognition. *International journal of computer vision*, **10**(3), June 1993.

[21] A. ALI, S. A. M. GILANI, N. A. MEMON, Affine Normalized Invariant Functionals Using Independent Component Analysis. 10*th* *IEEE International Multi-topic Conference* (INMIC 06), December 2006, pp. 94–99.

Received: February, 2007
Revised: February, 2008
Accepted: August, 2008

Contact address:
Asad Ali
GIK Institute of Engineering Sciences & Technology
Faculty of Computer Science & Engineering
Topi-23460, Swabi
NWFP
Pakistan
e-mail: asad_82@yahoo.com
aali@giki.edu.pk

Syed Asif Mahmood Gilani
GIK Institute of Engineering Sciences & Technology
Faculty of Computer Science & Engineering
Topi-23460, Swabi
NWFP
Pakistan
e-mail:asif@giki.edu.pk

Nisar Ahmed Memon
GIK Institute of Engineering Sciences & Technology
Faculty of Computer Science & Engineering
Topi-23460, Swabi
NWFP
Pakistan
e-mail: memon@giki.edu.pk

ASAD ALI holds a BE degree in software engineering and is currently a graduate student at the Faculty of Computer Science and Engineering, GIK Institute of Engineering Sciences & Technology. He is a recipient of National Engineering and Scientific Commission (NESCOM) fellowship for MS studies at GIKI. He won the 2nd prize in the All Pakistan software competition, Softcom in 2003, 1st prize in GIKITECH 2005 and the 2nd best paper award at IEEE International conference on Emerging Technologies in 2006. His areas of interest include invariant feature extraction, intelligent transportation systems, image segmentation, image denoising, machine cognition and spectral graph theroy.

SYED ASIF MEHMOOD GILANI holds a Ph.D degree in digital image processing with specialization in the field of digital image watermarking, especially in the data security issues. His research is focused on the study of digital image watermarking for copyright protection. In addition, he has also made significant contributions in data authentication and image steganography. He has been involved in teaching and research at undergraduate as well as postgraduate levels. Dr. Gilani has been awarded with the distinction in M.Sc, and Ph.D, Scholarship from the Greek State. He has published a number of research papers internationally. He is currently an Assistant Professor and Dean at the Faculty of Computer Science and Engineering, GIK Institute of Engineering Sciences & Technology.

NISAR AHMED MEMON received his BE(CS) from Mehran University of Engineering and Technology. At present he is Ph.D Scholar at the Faculty of Computer Science and Engineering, GIK Institute of Engineering Sciences and Technology, Pakistan. He is also a faculty member of Quaid-e-Awam University of Engineering Sciences and Technology, Pakistan. His research interests include medical image processing, digital watermarking of medical images, data authentication and security.