

Implicitization of Parametric Curves by Matrix Annihilation

Hulya Yalcin[†], Mustafa Unel^{†*}, William Wolovich[†]

[†]Division of Engineering, Brown University, RI

^{†*}Center for Computational Vision and Control, Yale University, CT

Abstract

Both parametric and implicit representations can be used to model 2D curves and 3D surfaces. Each has certain advantages compared to the other. Implicit polynomial (IP) methods are not as popular as parametric procedures because the lack of general procedures for obtaining IP models of higher degree has prevented their general use in many practical applications. In most cases today, parametric equations are used to model curves and surfaces. One such parametric representation, elliptic Fourier Descriptors (EFD) has been widely used to represent 2D and 3D curves, as well as 3D surfaces. Although EFDs can represent nearly all curves, it is often convenient to have an implicit algebraic description $F(x,y)=0$, especially for determining whether given points lie on the curve. Algebraic curves and surfaces also have proven very useful in many model-based applications. Various algebraic and geometric invariants obtained from these implicit models have been studied rather extensively. In this paper, we present a new non-symbolic implicitization technique called the matrix annihilation method, for converting parametric Fourier representations to implicit polynomial form.

1. INTRODUCTION

In general, any closed curve can be described in terms of a set of two (or three for space curves) Fourier series whose coefficients are called elliptic Fourier descriptors (EFDs). The advantages of using EFDs are that the shape information is concentrated in the low frequency parts [1-5], hence shape can be described by the first few coefficients. Invariants derived from EFDs have been used for identification [5,6]. Algebraic curves/surfaces have also proven to be very useful for shape representation [7,8]. In the past few years, implicit representations have been used more frequently, allowing a better treatment of several problems. One example is the point classification problem, which is easily solved with the implicit representation since it consists of a simple evaluation of the implicit functions. Invariants associated with these models have also been employed in several model-based vision and pattern recognition applications [9-13]. However, there is very little in the literature on higher degree IP models for large or entire free-form shapes because of the lack of tractable computational procedures for obtaining and analyzing such models. The implicit polynomial that defines a curve is not easily determined. Although several different fitting algorithms have been proposed thus far for directly determining IP models from point data sets, the computational cost of many of them is quite high because of the nonlinear optimization procedures required to obtain acceptable fits. Lack of stability of parameters has been another weakness. The problem of excessive number of parameters in implicit representations was first studied in [13]. The 3L fitting algorithm [14] exhibits significantly improved curve representation accuracy and stability. However, there is significant value to further improvement in coefficient stability to make algebraic curves more generally and robustly applicable

for object recognition. Instead of obtaining algebraic curve representations directly from points, it is also possible to convert parametric equations (which may be obtained from boundary points) to implicit ones. Sederberg has extensively studied the problem of converting parametrically defined curves to implicit form using elimination theory, which involves computing the determinant of Sylvester's matrix [15-17]. Later, Hong employed Sylvester's matrix –also known as the resultant method– for a particular class of problems involving trigonometric polynomials, taking advantage of their special structure [18]. A significant disadvantage of implicitization by Sylvester's matrix elimination method is that it involves computing the determinant of a matrix which contains symbolic variables. Ercil *et al* [19] studied the problem of converting between parametric and implicit forms based on polar/spherical coordinate representations. However, their technique is valid only for star-shaped objects.

In this paper, we present a new, non-symbolic implicitization technique, called the *matrix annihilation method*, for converting parametric Fourier representations to algebraic (implicit polynomial) representations. We should note that our method is numerical, so that we can obtain higher order polynomial curves than previously possible through implicitization. Furthermore, our procedure is computationally efficient. The structure of the paper is as follows: Section 2 reviews implicitization by Sylvester's matrix. In section 3, our approach for implicitization is described. Section 4 illustrates our procedure using several examples. Section 5 describes two vision-based examples, and some concluding remarks are given in Section 6.

2. Review of Implicitization by Sylvester's Matrix

Sylvester's matrix elimination method [17] can be used to implicitize parametric polynomial curves such as $x=a_2t^2+a_1t+a_0$ and $y=b_2t^2+b_1t+b_0$. In particular, if we re-write these equations as $a_2t^2+a_1t+(a_0-x)=0$ and $b_2t^2+b_1t+(b_0-y)=0$, the resultant of these two polynomials is then defined by the determinant of Sylvester's matrix, namely

$$S = \begin{bmatrix} a_2 & a_1 & a_0 - x & 0 \\ 0 & a_2 & a_1 & a_0 - x \\ b_2 & b_1 & b_0 - y & 0 \\ 0 & b_2 & b_1 & b_0 - y \end{bmatrix} \Rightarrow \begin{aligned} |S| &= b_2^2x^2 - 2a_1b_2xy + a_1^2y^2 \\ &+ (2a_2b_2b_0 + a_1b_1b_2 - a_1^2b_1^2 - 2a_1b_1^2)x \\ &+ (a_1b_1a_1 + 2a_2b_2a_0 - 2a_2^2b_0 - a_1^2b_2)y \\ &+ (a_1b_2 - a_1b_1)(a_1b_0 - a_0b_1) \\ &+ (a_1b_0 - a_0b_2)y = 0 \end{aligned}$$

an implicit algebraic curve that is equivalent to the parametric curve. In general, if x and y are polynomials of degree p , the corresponding algebraic curve also will have degree p . To further illustrate this procedure, consider the parametric curve defined by the equations: $x = 0.4 + 0.5 \cos t - 2 \sin 2t$ and

$y = 0.6 + 0.2 \sin t - 2 \sin 2t$. By substituting the following equivalent complex relations for $\cos kt$ and $\sin kt$

$$\cos kt = (e^{ikt} + e^{-ikt})/2, \quad \sin kt = (e^{ikt} - e^{-ikt})/2 \quad (2.1)$$

and then substituting $z = e^{it}$ and multiplying both equations by z^2 in order to make all z powers positive, we obtain

$$\begin{aligned} 0 &= iz^4 + 0.25z^3 + (0.4 - x)z^2 + 0.25z - i \\ 0 &= 0.35z^4 - 0.1iz^3 + (0.6 - y)z^2 + 0.1iz + 0.35 \end{aligned}$$

Using elimination theory, the determinant of Sylvester's matrix is then defined by

$$S = \begin{bmatrix} -i & 0.25 & 0.4-x & 0.25 & i & 0 & 0 & 0 \\ 0 & -i & 0.25 & 0.4-x & 0.25 & i & 0 & 0 \\ 0 & 0 & -i & 0.25 & 0.4-x & 0.25 & i & 0 \\ 0 & 0 & 0 & -i & 0.25 & 0.4-x & 0.25 & i \\ 0.35 & 0.1i & 0.6-y & -0.1i & 0.35 & 0 & 0 & 0 \\ 0 & 0.35 & 0.1i & 0.6-y & -0.1i & 0.35 & 0 & 0 \\ 0 & 0 & 0.35 & 0.1i & 0.6-y & -0.1i & 0.35 & 0 \\ 0 & 0 & 0 & 0.35 & 0.1i & 0.6-y & -0.1i & 0.35 \end{bmatrix},$$

which implies the (monic) quartic implicit polynomial curve

$$\begin{aligned} f_1(x, y) &= x^4 + 16.327x^2y^2 + 66.639y^4 - 1.6x^3 - 19.657x^2y - 13.06ky^2 \\ &- 16.035y^3 - 1.332x^2 + 15.726xy + 79.04y^2 + 1.578x + 24.698y - 3.736 = 0 \end{aligned}$$

that is shown in Figure 1(a).

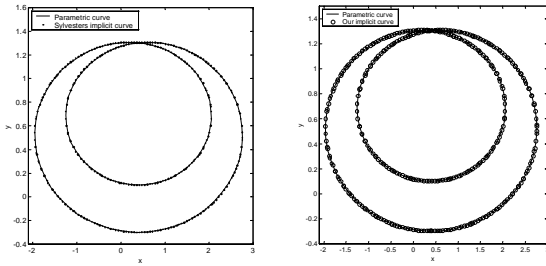


Figure 1 (a) Parametric curve and implicit curve obtained by Sylvester's elimination method, (b) parametric curve and implicit curve obtained by matrix annihilation method are superimposed.

3. Our Approach: Implicitization by Matrix Annihilation

Consider an n -harmonic elliptic Fourier descriptor representation of any 2-D curve, namely¹

$$\begin{aligned} x(t) &= a_0 + \sum_{k=1}^n (a_k \cos kt + b_k \sin kt) \\ y(t) &= c_0 + \sum_{k=1}^n (c_k \cos kt + d_k \sin kt) \end{aligned} \quad (3.1)$$

where (a_0, c_0) is the center of the curve. Using (2.1), we can express (3.1) as

$$\begin{aligned} x(t) &= A_0 + \sum_{k=1}^n A_k e^{ikt} + B_k e^{-ikt} \\ y(t) &= C_0 + \sum_{k=1}^n C_k e^{ikt} + D_k e^{-ikt} \end{aligned} \quad (3.2)$$

¹ The coefficients of $\cos kt$ and $\sin kt$ in (3.1) can be uniquely determined from an ordered sequence of points which describe the boundary of a curve using the relations given in Chapter 2 of [6]. We will assume here that this operation has been performed and, therefore, that the Fourier coefficients are known.

where $A_k = (a_k - ib_k)/2$, $B_k = (a_k + ib_k)/2$, $C_k = (c_k - id_k)/2$, $D_k = (c_k + id_k)/2$ for $k=1, \dots, n$, with $A_0 = a_0$ and $C_0 = c_0$. These equations can be expressed in a more compact form by substituting z for e^{it} ,

$$\begin{aligned} x(z) &= A_0 + \sum_{k=1}^n (A_k z^k + B_k z^{-k}) \equiv \sum_{k=-n}^n g[k] z^k \\ y(z) &= C_0 + \sum_{k=1}^n (C_k z^k + D_k z^{-k}) \equiv \sum_{k=-n}^n h[k] z^k \end{aligned} \quad (3.3)$$

where

$$g[k] = \begin{cases} A_k & \text{if } k > 0 \\ A_0 & \text{if } k = 0 \\ B_k & \text{if } k < 0 \end{cases} \quad h[k] = \begin{cases} C_k & \text{if } k > 0 \\ C_0 & \text{if } k = 0 \\ D_k & \text{if } k < 0 \end{cases}$$

If the g and h sequences are written as vectors $g = [B_n \dots B_1 A_0 A_1 \dots A_n]$ and $h = [D_n \dots D_1 C_0 C_1 \dots C_n]$, (3.3) can then be re-written as $x(z) = g \cdot \bar{z}$ and $y(z) = h \cdot \bar{z}$ where $\bar{z}^T = [z^{-n} \dots z^{-1} 1 z \dots z^n]$. We next employ the time convolution property of the z -transform: "if $g[k] \leftrightarrow x(z)$ and $h[k] \leftrightarrow y(z)$ then $g[k] * h[k] \leftrightarrow x(z)y(z)$ ". Note that convolution in discrete-time domain corresponds to multiplication in the z -domain. For example, $x^2 = x(z)x(z) = Z\{g[k] * g[k]\}$, $xy = x(z)y(z) = Z\{g[k] * h[k]\}$, $y^2 = y(z)y(z) = Z\{h[k] * h[k]\}$. The monomials $x^p y^q$ for different p and q values can be found similarly. We can therefore write

$$\begin{bmatrix} 1 \\ x \\ y \\ x^2 \\ xy \\ y^2 \\ x^3 \\ \vdots \\ xy^{d-1} \\ y^d \end{bmatrix} = \begin{bmatrix} 0 \dots 0 & 1 & 0 \dots 0 \\ & g & \\ & h & \\ & g * g & \\ & g * h & \\ & h * h & \\ & g * g * g & \\ & \vdots & \\ & \underbrace{g * h * \dots * h}_{d-1} & \\ & \underbrace{h * h * \dots * h}_d & \end{bmatrix} \begin{bmatrix} z^{-nd} \\ \vdots \\ z^{-1} \\ 1 \\ z \\ \vdots \\ z^d \end{bmatrix} \quad (3.4)$$

P : Convolution Matrix

or simply $\Gamma = P\bar{z}$ for some complex $(d+1)(d+2)/2 \times (2dn+1)$ matrix P . We next re-write P as

$$P = \hat{P} \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ i & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & i & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & 0 & i \end{bmatrix}$$

for some unique, real $(d+1)(d+2)/2 \times (2d^2+1)$ matrix \hat{P} . We then determine the "largest" $(d+1)(d+2)/2 - 1 = d(d+3)/2$ columns of \hat{P} via an orthogonal-triangular decomposition

defined by $QR = \hat{P}E$, where Q is a unitary matrix, R is an upper triangular matrix whose diagonal elements are of decreasing absolute value, and E is a permutation matrix which orders the columns of $\hat{P}E$ in correspondence with those of QR . The unique unit vector v that annihilates the first $d(d+3)/2$ columns of $\hat{P}E$, which we will define as \tilde{P} , then yields an appropriate (non-monic) implicit polynomial function as the product $v \Gamma = f_d(x, y) = 0$. The Matlab routines “qr” and “null” are used to perform the required computations. Implicitization of truncated Fourier descriptors with n harmonics implies an algebraic equation of degree $d=2n$.

4. SOME EXAMPLES

In this section, we present the implicitization of several curves, some with strong discontinuities and some self-intersecting. Given an ordered sequence of points on each curve, we first computed the parametric Fourier coefficients of the curve. We then implicitized the curves both by our matrix annihilation method and Sylvester’s elimination method. Both implicitization methods were implemented in Matlab. For Sylvester’s elimination method, we used the Symbolic Toolbox of Matlab and/or Maple. Assuming that n harmonic Fourier coefficients are used to represent the curve, our method involves computing the annihilating vector of \tilde{P} , whereas implicitization by Sylvester’s elimination method involves computing the determinant of a $4n \times 4n$ symbolic matrix. As the number of harmonics used to represent curves increase, the computation time increases for both methods. However, since Sylvester’s elimination method involves symbolic computation, the computation time increases faster than our method, as we will show.

4.1. Convert 2 harmonic EFD curve to 4th degree IP curve:

To illustrate our matrix annihilation procedure, consider the earlier parametric curve defined by

$$x = 0.4 + 0.5 \cos t - 2 \sin 2t, \quad y = 0.6 + 0.2 \sin t + 0.7 \cos 2t$$

whose Fourier coefficients are given by $a_0 = 0.4, c_0 = 0.6$

$$\begin{bmatrix} a_1 & a_2 \end{bmatrix} = \begin{bmatrix} 0.5 & 0 \end{bmatrix} \quad \begin{bmatrix} b_1 & b_2 \end{bmatrix} = \begin{bmatrix} 0 & -2 \end{bmatrix} \\ \begin{bmatrix} c_1 & c_2 \end{bmatrix} = \begin{bmatrix} 0 & 0.7 \end{bmatrix} \quad \begin{bmatrix} d_1 & d_2 \end{bmatrix} = \begin{bmatrix} 0.2 & 0 \end{bmatrix}$$

The complex Fourier coefficients are then determined to be $A_0 = 0.4, C_0 = 0.6, [A_1 \ A_2] = [0.25 \ i], [B_1 \ B_2] = [0.25 \ -i], [C_1 \ C_2] = [-0.1i \ 0.35], [D_1 \ D_2] = [0.1i \ 0.35]$. The g and h sequences are then defined as

$$g = [-i \ 0.25 \ 0.4 \ 0.25 \ i], \quad h = [0.35 \ 0.1i \ 0.6 \ -0.1i \ 0.35]$$

which subsequently imply (3.4). The (15×14) \tilde{P} matrix is then determined as outlined above and the vector that annihilates this matrix is found to be

$$v = \begin{bmatrix} -0.0191 & 0.0081 & 0.1265 & -0.0068 & 0.0806 & 0.4049 & -0.0082 \\ -0.1007 & -0.0669 & -0.8215 & 0.0051 & 0 & 0.0836 & 0 & 0.3414 \end{bmatrix}$$

Reordering the terms in lexicographic order and dividing by the leading coefficient, we obtain the following monic implicit polynomial curve

$$f_4(x, y) = x^4 + 16.327x^2y^2 + 66.639y^4 - 1.6x^3 - 19.657x^2y - 13.06xy^2 - 160.35y^3 - 1.332x^3 + 15.726xy + 79.04y^2 + 1.578x + 24.698y - 3.736 = 0$$

that is shown in Figure 1(b). The annihilated matrix \tilde{P} defined by our method is 15×14 , and it takes 0.06 seconds to implicitize. Sylvester’s matrix is 8×8 , but it takes 7.25 seconds to implicitize.

4.2. Convert 3 harmonic EFD curve to 6th degree IP curve:

The parametric curve in Figure 2(a) is represented by the following Fourier coefficients:

$$a_0 = -0.0906, \quad c_0 = -0.1194 \\ \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix} = \begin{bmatrix} -0.1827 & -0.1902 & -0.1302 \end{bmatrix} \\ \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix} = \begin{bmatrix} -0.3267 & -0.1753 & -0.1568 \end{bmatrix} \\ \begin{bmatrix} c_1 & c_2 & c_3 \end{bmatrix} = \begin{bmatrix} -0.3383 & -0.153 & -0.0971 \end{bmatrix} \\ \begin{bmatrix} d_1 & d_2 & d_3 \end{bmatrix} = \begin{bmatrix} 0.3385 & 0.0992 & -0.1589 \end{bmatrix}$$

The 6th degree IP obtained by both our annihilation method and Sylvester’s elimination method is the same, namely

$$f(x, y) = x^6 - 6.4992x^5y + 17.6741x^4y^2 - 25.7403x^3y^3 + 21.1742x^2y^4 - 9.3283xy^5 + 1.7195y^6 - 2.5056x^5 + 10.5328x^4y - 16.0141x^3y^2 + 10.0422x^2y^3 - 1.6236xy^4 - 0.4356y^5 + 0.1532x^4 + 3.6857x^3y - 7.8187x^2y^2 + 4.8756xy^3 - 0.7753y^4 + 2.1925x^3 - 2.9669x^2y + 0.8362xy^2 + 0.0254y^3 + 0.6077x^2 - 0.5863xy + 0.0803y^2 - 0.0041x + 0.0599y - 0.0096 = 0$$

and shown in Figure 2(a). The annihilated matrix \tilde{P} defined by our method is 28×27 and it takes 0.06 seconds to implicitize. Sylvester’s matrix is 12×12 , but it takes 8.46 seconds to implicitize.

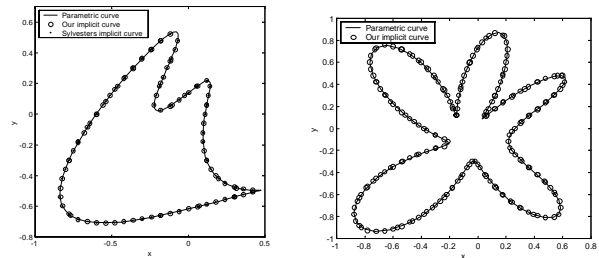


Figure 2 (a) Parametric curve, implicit curve (circles) obtained by matrix annihilation and implicit curve (dots) obtained by Sylvester’s elimination methods are superimposed. (b) Parametric curve and implicit curve (circles) obtained by matrix annihilation method are superimposed.

4.3. Convert 6 harmonic EFD curve to 12th degree IP curve:

The parametric curve in Figure 2(b) is represented by following Fourier coefficients:

$$a_0 = -0.057, \quad c_0 = -0.0157 \\ \begin{bmatrix} a_1, a_2, a_3, a_4, a_5, a_6 \end{bmatrix} = \begin{bmatrix} -.5224, -.0574, 0.0830, 0.1513, 0.0262, 0.1703 \end{bmatrix} \\ \begin{bmatrix} b_1, b_2, b_3, b_4, b_5, b_6 \end{bmatrix} = \begin{bmatrix} -.09, 0.089, 0.0258, -.0154, 0.0251, -.0113 \end{bmatrix} \\ \begin{bmatrix} c_1, c_2, c_3, c_4, c_5, c_6 \end{bmatrix} = \begin{bmatrix} -.1396, 0.0652, 0.0682, 0.0843, -.0832, 0.0176 \end{bmatrix} \\ \begin{bmatrix} d_1, d_2, d_3, d_4, d_5, d_6 \end{bmatrix} = \begin{bmatrix} 0.6211, 0.0893, 0.0938, 0.1858, -.0864, -.1864 \end{bmatrix}$$

The 12th degree implicit polynomial² obtained by our annihilation method is shown in Figure 2(b). The annihilated

² The coefficients of the implicit polynomial equations which define our remaining curves can be obtained from the authors.

matrix \tilde{P} defined by our method is 91x90 and it takes 1.21 seconds to implicitize. When the number of harmonics used to represent a curve is greater than 4 (degree of target implicit polynomial greater than 8), we found that implicitization by Sylvester's elimination method is impossible using Matlab.

5. Extracting Shape Features from IP Models

Once an implicit polynomial that defines the shape of an object has been determined, it can be used for object recognition in a variety of ways. As shown in [12-14], algebraic invariants can be defined and determined directly from implicit polynomial models. These invariants can then be used to compare and identify similar objects. Implicit polynomial representations can also be used to characterize the local geometry of a curve, i.e. curvature. The classical approach in curvature computation is to use local curve fitting or structural models for a local neighborhood of the curve, which typically perform poorly in the vicinity of singularities. To avoid this problem, we use the implicit representation obtained by our matrix annihilation method to compute curvature of a curve and determine whether the objects are the same or similar.

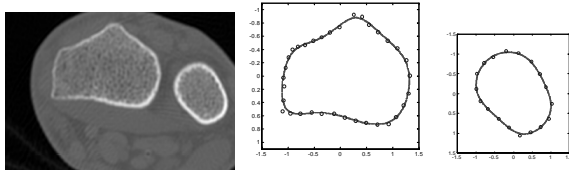


Figure 3 (a) Carpal3d_01_01 image. (b) Contour points of the first bone (circle) and 10th degree IP curve obtained by our matrix annihilation (solid). (c) Contour points of the second bone (circle) and 8th degree IP curve (solid).

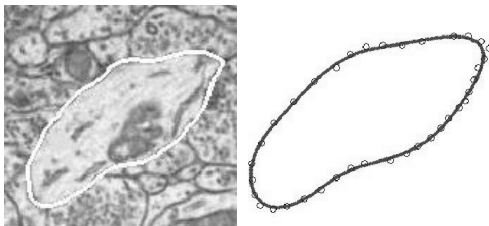


Figure 4 (a) Cell image, (b) contour points of the cell (circle) and 10th degree IP (5 harmonic) (solid).

Consider the carpal bone image depicted in Figure 3 (a). Using our matrix annihilation algorithm, a 10th degree IP curve is determined that describes the boundary of the first bone and an 8th degree IP curve is determined that describes the boundary of the second bone, which is depicted in Figure 3 (b) and (c) respectively. Similarly, implicit representation of the cell in Figure 4(a) is shown in Figure 4 (b). Computing the curvature of carpal bone shapes from their implicit representation together with perpendicular distance computation, we aim to use such information to quantify the degree to which the shape of a particular carpal bone differs from that of an ideal one.

6. CONCLUSIONS

Implicit polynomial representations are very useful for modeling given point data sets, and numerous papers have been written which illustrate their importance in image understanding and

object recognition. A variety of methods have been devised for directly fitting given data sets to implicit polynomial curves and, although such methods are continually being refined and improved, alternative implicitization procedures are equally important and useful. In this report, we have demonstrated a new matrix annihilation method for efficiently converting curves defined by elliptic Fourier descriptors to algebraic ones. As we have explicitly illustrated, our matrix annihilation method works very well and efficiently in many higher order cases where symbolic-based methods fail. We are currently working on extending our matrix annihilation method to implicitize both 2D curves and 3D surfaces that are defined by a variety of parametric representations. We also are investigating the use of these implicit models in a variety of computer vision applications. The two simple examples in Section 5 were given to illustrate the potential utility of our method, and spatial limitations prevent us from presenting additional computer vision applications here.

7. REFERENCES

- [1] T.P. Wallace and O.R. Mitchell, "Analysis of Three-Dimensional Movement Using Fourier Descriptors" *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.2, no.6, pp. 583-588, 1980.
- [2] F.P. Kuhl and C.R. Giardina, "Elliptic Fourier Features of a Closed Contour," *Computer Graphics and Image Processing*, vol.18, 1982.
- [3] C.T. Zahn and R.Z. Roskies, "Fourier Descriptors for Plane Closed Curve," *IEEE Trans. Computers*, vol. 21, no. 3, pp. 269-281,1972.
- [4] G.H. Granlund, "Fourier Preprocessing for Hand Print Character Recognition," *IEEE Trans. Computers*, vol. 21, pp. 195-201, 1972.
- [5] C.S. Lin and C.L. Hwang, "New Forms of Shape Invariants From Elliptic Fourier Descriptors," *Pattern Recognition*, 20(5):535-545, 1987.
- [6] P.E. Lestrel, *Fourier Descriptors and Their Applications in Biology*, Cambridge Univ. Press, 1997.
- [7]G.Taubin, F.Cukierman, S.Sullivan, J.Ponce and D.J. Kriegman, "Parameterized Families of Polynomials for Bounded Algebraic Curve and Surface Fitting," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 16(3):287-303, March 1994.
- [8] D. Keren, D. Cooper and J. Subrahmonia, "Describing Complicated Objects by Implicit Polynomials," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 16, pp. 38-53, 1994.
- [9] W. A. Wolovich, and Mustafa Unel, "The Determination of Implicit Polynomial Canonical Curves," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20 (8), 1998.
- [10] M. Unel, and W. A. Wolovich, "Pose Estimation and Object Identification Using Complex Algebraic Representations," *Pattern Analysis and Applications Journal*, Vol. 1 (3), 1998.
- [11] M. Unel and W. A. Wolovich, "On the Construction of Complete Sets of Geometric Invariants for Algebraic Curves," *Advances in Applied Mathematics* 24, 65-87, 2000.
- [12] M. Unel and W. A. Wolovich, "A New Representation for Quartic Curves and Complete Sets of Geometric Invariants," *Int. Jour. of Pattern Recognition and Artificial Intelligence*, 13 (8), 1999.
- [13] J. Subrahmonia, D. B. Cooper, and D. Keren, "Practical Reliable Bayesian Recognition of 2D and 3D Objects using Implicit Polynomials and Algebraic Invariants," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(5):505-519, 1996.
- [14] M.M. Blane, Z. Lei, H. Civi and D. B. Cooper, "The 3L Algorithm for Fitting Implicit Polynomial Curves and Surfaces to Data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(3), 2000.
- [16] T.W.Sederberg, "Implicit and Parametric Curves and Surfaces for Computer Aided Geometric Design," Ph.D. Thesis, Dept. Mechanical Engineering, Purdue University, 1983.
- [17] T.W.Sederberg and D.C.Anderson, "Implicit Representation of Parametric curves and Surfaces," *Comput. Vision&Graph. Image Process.*, 28(1):72-84, 1984.
- [18] T. W. Sederberg and R.N. Goldman, "Algebraic geometry for Computer-aided Geometric Design," *IEEE Comput. Graph. Appl.* 6(6), 52-59, 1986.
- [19] H. Hong, "Implicitization of Curves Parametrized by Generalized Trigonometric Polynomials," *Proceedings of Applied Algebra, Algebraic Algorithms and Error Correcting Codes (AAECC-11)*, pp. 285-296, 1995.
- [20] C. Unsalan and A. Ercil, "Conversions between Parametric and Implicit Forms Using Polar/Spherical Coordinate Representations," *Computer Vision and Image Understanding* 81, 1-25 (2001).