Combining Implicit Polynomials and Geometric Features for Hand Recognition

Cenker Oden[1], Aytul Ercil[2], Burak Buke[1]
[1]*Bogazici University, Alper Atalay BUPAM Laboratory, 80815, Bebek / Istanbul TURKEY*
{oden,bukeb}@boun.edu.tr
[2]*Sabanci University, VPA Laboratory, 81474 Tuzla / Istanbul TURKEY*
aytulercil@sabanciuniv.edu.tr

Keywords: Hand recognition, Implicit polynomials, algebraic invariants

Abstract

Person identification and verification using biometric methods is getting more and more important in today's information society. Hand recognition is a promising biometric because of its simplicity and fairly good performance of verification. In this work, implicit polynomials, which have proven to be very successful in object modeling and recognition, have been proposed for recognizing hand shapes and the results are compared with existing methods.

1. Introduction

As the personal and institutional security requirements increase, a person has to remember lots of passwords, pin numbers, account numbers, voice mail access numbers and other security codes. In the future, biometric systems will take the place of this concept since it is more convenient and reliable. Trying to come up with an all-purpose, or at least multi-purpose, personal identifier is what the art and science of biometrics is all about. A broad variety of physical characteristics are now being tested to determine the potential accuracy and ultimate consumer acceptance of their biometric measurement as personal identification standards. Up to now, biometric properties like fingerprint, face, voice, handwriting and iris were the subjects of many research efforts and used in different types of identification and verification systems. But the main reason of increased interest in this research area is that as the technology develops, these kinds of systems are more likely to run on the personal devices such as mobile phones and laptops [3].

In many access control systems like border control, personnel follow-up, important point is verification rather than identification. Hand recognition systems are very appropriate for these purposes, because they do not cause anxiety for the user like fingerprint and iris systems. People's ease of acceptance due to their convenience, and their easy and cheap setup are the major superiorities of hand geometry based recognition systems to other systems.

There are some commercial hand recognition systems available, and new algorithms were proposed recently giving better performance. Most of these methods rely mainly on geometric features and use the same logic for feature extraction. In this paper, we propose a new method for recognizing hand shapes using implicit polynomials. The performance of the proposed system will be compared with existing methods and a way to combining geometric features with the invariants calculated from implicit polynomial fits of the hand will be studied. Our work also offers an improvement in terms of user friendliness by giving a greater freedom to the user for placing his/her hand on the platform; because we do not use fixation pegs, unlike most of the former methods.

2. Implicit Polynomials

Implicit polynomial 2D curves and 3D surfaces are potentially among the most useful object and data representations for use in computer vision and image analysis because of their interpolation property, Euclidean and affine invariants, as well as their ability to represent complicated objects. There have been great improvements concerning implicit polynomials with its increased use during the late 80's and early 90's [2, 5, 10]. Recently, new robust and consistent fitting methods like 3L fitting, gradient-one fitting, Fourier fitting have been introduced [6,11], making them feasible for real-time applications for object recognition tasks.

The implicit polynomial model is given as:

$$f_n(x,y) = \sum_{0 \le i,\, j;\, i+j \le n} a_{ij} x^i y^j =$$

$$a_{00} + a_{10}x + a_{01}y + \ldots + a_{n0}x^n + a_{n-1,1}x^{n-1}y + \ldots + a_{0n}y^n = 0$$

An implicit polynomial can be completely characterized by its coefficient vector:

$$\lfloor a_{n0}, a_{n-1,1}, a_{n-2,2}, \ldots a_{0,n-2}, \ldots a_{10}, a_{01}, a_{00} \rfloor \Leftrightarrow f_n(x,y)$$

An implicit polynomial curve is said to represent an object

$$\Gamma_0 = \{(x_i, y_i) | i = 1, ..., K\}$$

if every point of the shape Γ_0 is in the zero set of the implicit polynomial

$$Z(f) = \{(x, y) | f(x, y) = 0\}$$

The zero set of an implicit polynomial fitted to the data will usually be close to the points of Γ_0 but cannot contain all of them.

The most important and the fundamental problem in implicit polynomials is the fitting problem, namely to find the implicit polynomial function f(x,y), or the corresponding coefficient vector that best represents the object. The fitting procedure has to be robust, which means a small amount of change in the data should not cause a relatively huge amount of change in the coefficients of the fitted implicit polynomial. Traditional fitting methods such as least squares fitting lacked this property, and the slightest change in data set caused dramatic differences in resulting coefficients. New algorithms such as 3L fitting [6], gradient-one fitting [9] and Fourier fitting [11] have this desirable character, enabling us to use implicit polynomials for object recognition tasks in a reliable manner.

The main advantage of implicit polynomials for recognition is the existence of algebraic and geometric invariants, which are functions of the polynomial coefficients that do not change after a coordinate transformation. The algebraic invariants that are found by Civi [2] and Keren [4] are global invariants and are expressed as simple explicit functions of the coefficients. Another set of invariants that have been mentioned by Wolovich et al. is derived from the covariant conic decompositions of implicit polynomials [12]. Their performance have been tested with different complicated objects and they were found to be very successful in object recognition tasks even in the presence of noise, and missing data [14].

3. Methodology

Despite the fact that commercial systems for hand recognition exist in the market, there aren't many and detailed studies on this field in the literature. However due to the reasons explained above, new methods have been proposed recently [1,3,8]. All of the methods proposed use various geometric features of hand (width height and length of the fingers, hand size, height profile, etc.). In our study, we tried to improve the success of the former methods by using implicit polynomials to model the fingers.

Initially, preliminary work was performed on the sample database that has been downloaded from [1], (including eight images from nine persons) and we tried our algorithm on these images, which gave a 98% of success in identification, encouraging us for future work. We then formed our own hand database by taking 40 images from 35 people. We used backlighting in order to take robust images independent of lighting conditions. We did not use the height profile of hands and constrained ourselves only to a single view of the hand. We did not also utilize fixation pegs as the methods we referenced employ; and did not constrain users; the only requirements were to place hands in the backlighted area and not to combine fingers. Our main motivation in doing so was the high performance of algebraic invariants independent of scaling and rotation. In this way, we were hoping to implement a system that is easier to use and more user friendly. For preprocessing, a LoG edge detector was applied to the acquired images and then images were enhanced to obtain a single-pixel-width boundary of the hand. 20 of these 40 images were used for training and the rest for test purposes. The image acquired from the prototype system and its processed output is seen Fig. 1 and Fig. 2.

Fig. 1. Original image taken with the setup

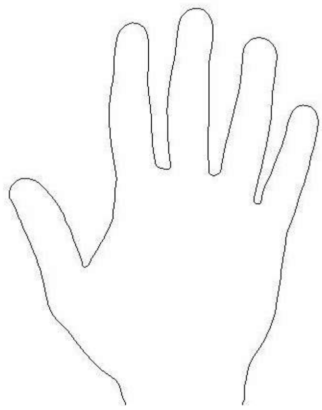

Fig. 2. Processed boundary image

Geometric features are calculated as seen in Fig. 3 [1,3]. Using the boundary data, a signature analysis was performed to find appropriate points and reliably extract the fingers. When calculating the signature, reference point was chosen as x=(first moment of the x coordinates of the boundary pixels), y=(height of the image), instead of center of mass of the boundary points. This specific choice resulted in a 1-D signal, from which five fingertips and four interfinger points could easily be found. A moving average filter was used to seek for local minima and maxima on this signal, which are the points we are looking for. The filter checks k points before and after the candidate point to see if averages of these distances are smaller or larger than the candidate point. When k was chosen greater than 30, we were able to find the fingertips and inter-finger points reliably. Using these points, we could both calculate the geometric features (by utilizing simple analytic geometry rules) and extract each finger's data, so that we could apply implicit polynomial fitting. Fourth and sixth degree implicit polynomials were then fitted to each of the fingers both using gradient-one fitting and 3L fitting. Resulting coefficient vectors were used to calculate Keren's and Civi's invariants for fourth degree fitting, and absolute geometric invariants for sixth degree fitting. In the next few subsections, we will review the different fitting techniques and the different invariants that are used throughout the study.

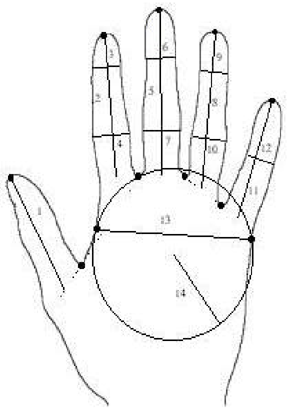

Fig. 3. Geometric features used to construct the feature vector illustrated.

3.1 FittingImplicit Polynomials to Data

An essential requirement for practicality of implicit polynomial techniques is to have robust and consistent implicit polynomial fits to data sets. The fitting problem of implicit polynomials is to find the implicit polynomial function f(x,y) that best represents the object. Since the implicit polynomial coefficients we find from the fittings are used to compute algebraic invariants or conic factor centers, which are used as primary geometric descriptors of the objects modeled in the recognition system, the fitting procedure has to be robust. This means a small amount of change in the data should not cause a relatively huge amount of change in the coefficients of the fitted implicit polynomial. In that sense, some researchers claim that the stability of the fitting is more important than the goodness of the fitting itself. In our work, we used the 3L fitting technique, which has a good record of robustness when compared with other methods in the literature. [6, 9] 3L fitting uses a pair of level sets Γ_{+c} and Γ_{-c} , where both level set are at a distance +c (outside the data set) and -c (inside the data set) from the data set, respectively. [6] Besides the original data set, the 3L algorithm uses a pair of synthetically generated data sets consisting of points from Γ_{+c} and Γ_{-c}. This procedure forces the solution to be more stable around the data.

The gradient one algorithm is based on controlling the value of the first derivatives along the zero set, i.e, the gradient of the 2D polynomial.

By definition, the gradient vector along the zero set curve of the polynomial is always perpendicular to this curve. Thus, if we can compute the local tangent to the data curve at each point of the data set, we force the implicit polynomial gradient to be perpendicular to the local tangent and with unit norm. This will force the zero set of the polynomial to respect the local continuity of the data set. [9]

3.2. Algebraic Invariants using the Symbolic Method

Symbolic method represents the invariants of a system of polynomials as products of formal bracket symbols in dummy variables, formed according to a simple system of rules. Monomials in the dummy variables are replaced by coefficients of the original polynomials, yielding the invariants. A significant result is that all algebraic invariants ultimately involve determinants. Implementation of the algorithm was done by the REDUCE symbolic computation software. The two inputs of the algorithm are the degree of the form and the maximum degree for the invariants sought. We refer the reader to [9] for the details of the algorithm.

Some affine invariants found by Civi using the symbolic method for fourth degree implicit polynomials are given in [2].

If an invariant is to be used in object recognition under affine and projective transformation of the image plane, it should be an absolute weight invariant and for the Euclidean case every invariant is automatically an absolute weight invariant.

To find the absolute invariants, we have to both take weight and degree of invariants into consideration. Below are the two invariants, which are absolute invariants, obtained from I_1, I_2, I_3, I_4.

$$\bar{I}_1 = \frac{I_1 I_4}{I_2 I_3} \qquad (3.2.1)$$

$$\bar{I}_2 = \frac{I_1^2}{I_3^2 I_4} \qquad (3.2.2)$$

The other invariants obtained using the symbolic method are given in the Experiments section.

3.3. Invariant Computation by Keren

If a particular structure $I = \sum_{0 \le i \le j \le N} y_{ij} a_i a_j$ is assumed to be an invariant, where $\{ a_i \}^{i=N}_{i=1}$ are the coefficients of the implicit polynomial, then an invariant expression of transformed coefficients, which is $\bar{I} = \sum_{0 \le i \le j \le N} y_{ij} \bar{a}_i \bar{a}_j$ has to be equal to or very near in value to I, since it is an invariant. In the case that the transformation T is a rotation with parameter q , each $\bar{a}_i$ is a function of the a_i's and q . If we denote the relation between the a_i's and q and the $\bar{a}_i$'s by f , then formally f is: $f : \Re x \Re^N \to \Re^N$.

For $I = \sum_{0 \le i \le j \le N} y_{ij} a_i a_j$ to be an invariant, the following has to hold for every coefficient vector $a = \sum_{i=1}^{N} a_i$ and every angle q :

$$y\left(\{a_i\}\right) = \sum_{0 \le i \le j \le N} y_{ij} a_i a_j = \sum_{0 \le i \le j \le N} y_{ij} \bar{a}_i \bar{a}_j = y\left(\{\bar{a}_i\}\right) \quad (3.3.1)$$

For Equation 3.2.1 to hold, it is necessary and sufficient that for every $a = \sum_{i=1}^{N} a_i$,

$$\left(\frac{\P}{\P q} y\left[f\left(q, p\right)\right] \right)_{q=0} = 0 \qquad (3.3.2)$$

The proof of this theorem and the details of the algorithm can be found in [5].

The invariant found with this method are Euclidean invariants and are used as features in the recognition system we use. Some of the invariants found with this method are given below and the rest can be found in [5].

$$\begin{aligned} inv_1 &= 3a_{13}^2 - 8a_{04}a_{22} + 2a_{13}a_{31} + 3a_{31}^2 - 32a_{40}a_{04} - 8a_{22}a_{40} \\ inv_2 &= 3a_{04}^2 + 2a_{04}a_{22} + a_{13}a_{31} + 2a_{04}a_{40} + 2a_{22}a_{40} + 3a_{40}^2 \\ inv_3 &= a_{22}^2 - 3a_{13}a_{31} + 12a_{04}a_{40} \end{aligned} \quad (3.2.3)$$

3.4 Absolute Quantic Invariants with Conic Line Decomposition

Affine equivalence of two monic nth degree polynomials $f_n(x,y)$ and $\bar{f}_n(\bar{x},\bar{y})$ is defined as

$$f_n(x,y) = 0 \xrightarrow{A} f_n(ax + by + t_x, cx + dy + t_y) = s\bar{f}_n(\bar{x},\bar{y}) = 0 \quad (3.4.1)$$

The corresponding related points $\{x_m, y_m\}$ and $\{\bar{x}_m, \bar{y}_m\}$ of these two affine equivalent curves is defined by the condition that

$$\begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} = A \begin{bmatrix} \bar{x}_m \\ \bar{y}_m \\ 1 \end{bmatrix} \qquad (3.4.2)$$

where A is an affine transformation matrix.

Any two implicit polynomial of degree n, as defined by implicit polynomials $f_n(x,y)$ and $\bar{f}_n(x,y) = \sum_{r=0}^{n} \bar{h}_r(x,y)$ will be called quantic equivalent if

$$f_n(x,y) = \sum_{r=0}^{n} h_r(x,y) = 0 \xrightarrow{T} (h_0 / \bar{h}_0) \times \bar{f}_n(x,y) = \sum_{r=0}^{n} \frac{h_0 \bar{h}_r(x,y)}{\bar{h}_0} = 0 \quad (3.4.3)$$

A theorem [13] states that, if $\{x_m, y_m\}$ and $\{\bar{x}_m, \bar{y}_m\}$ are any two corresponding related-points of $f_n(x,y) = 0$ and $f_n(\bar{x},\bar{y}) = 0$ and $f_n'(x',y') = 0$ and $f_n''(x'',y'') = 0$ are their related point centered curves, then $f_n(x,y) = 0$ and $f_n(\bar{x},\bar{y}) = 0$ will be affine equivalent if and only if $f_n'(x',y') = 0$ and $f_n''(x'',y'') = 0$ are quantic equivalent. The proof of this theorem can be found in [13]. In the light of this theorem, we can first translate each curve to an equivalent curve with a corresponding related point at $\{0,0\}$, then determine the affine equivalence of the original curves by using the quantic equivalence of the related-point centered curves.

The related points we used in this paper are found by decomposing an implicit polynomial equation to find its conic factor centers and line factor intersections. The biggest advantage of this method is that we can find these points for higher degree polynomials such as 6th, 8th or even higher up to 18th degree. Hence we can find their corresponding absolute quantic invariants to test quantic equivalence which in turn implies the affine equivalence of the original curves.

To find the related points, as formulated in [10] we first rewrite the polynomial as the product of n/2 conics, plus a polynomial of degree n-2, namely

$$f_n(x,y) = \prod_{i=1}^{p} C_i(x,y) \prod_{j=1}^{n-2p} L_j(x,y) + f_{n-2}(x,y) \qquad (3.4.4)$$

In the above expression each $C_i(x,y)$ corresponds to a conic factor and each $L_j(x,y)$ corresponds to a line factor intersection. The conic factor centers and the line factor intersections defined in [10] are the related points we use in our work.

To determine the quantic equivalence of the related-point centered curves, we have to first determine the absolute quantic invariants accompanying them. The quantic invariants we used are Catalecticant of the quantic, discriminant of the quantic and the second degree quantic invariant introduced by Salmon. The number of absolute quantic invariants increases as we increase the degree of the fit because of the increasing number of quantics with higher degree fits.

As previously stated, one of the most useful properties of implicit polynomials is their interpolation property for locally missing data; and we employed this fact during our application as the extracted boundary data for fingers were not connected. Fig. 4 shows how implicit polynomials handle this situation.

4. Experiments

Throughout the experiments, Mahalanobis distance was used for classification purposes. Maximum interclass distance multiplied by 10 was used as a threshold for verification. This value was determined experimentally.

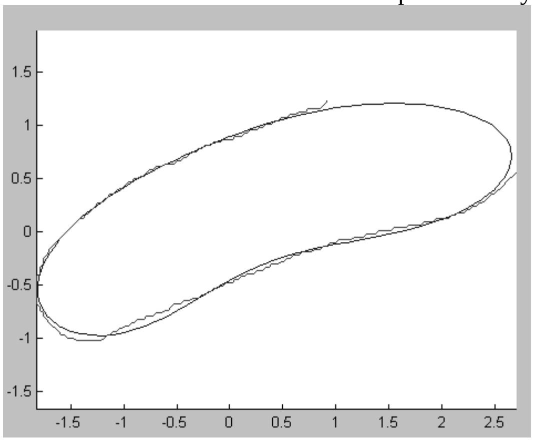

Fig. 4. A 4th degree fit to one of the fingers extracted from boundary data *(closed curve shows the fitted polynomial)*.

In the fourth degree case, we observed that Keren's invariants gave better results for all cases, and also gradient-one fitting slightly over performed 3L fitting. Sixth degree polynomials, which have higher degree of freedom, thus a better capability to model our data, surprisingly performed poorly compared to fourth degree models. This decrease in success rates can be explained by the fact that fingers are simpler objects, which can be fitted closely enough by fourth degree polynomials. Sixth degree polynomials do not necessarily model our data better; often the finger is again modeled by a fourth order curve, and there is another unbounded conic nearby, causing unstable results. (See Figure 5) Also, sixth degree polynomials require more parameters to be estimated, and with limited amount of data available, this may decrease the accuracy of the estimated parameters.

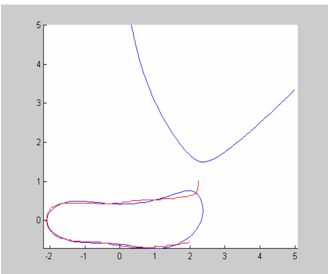

Fig. 5. A 6th degree fit to one of the fingers extracted from boundary data *(closed curve shows the fitted polynomial)*.

We wanted to combine geometric features and algebraic invariants in order to search for a better performance. As we had 20 training images, we could accomplish this goal in two ways, either selecting a subset of the features using some heuristics or using dimension reduction techniques like Principal Component Analysis. We tried both methods; for selecting features manually, we first eliminated some features since it was observed that they were misleading (e.g. some features for thumb since its geometry is not stable), as a consequence of the image taking policy, where minimum constraints were imposed on the user. With these subset of features, we achieved a recognition rate of 95%, and an identification rate of 99%.

Applying PCA to our 24-feature vector set resulted in 5 dimensions, retaining 99% of the original variance. Experiments that have been done in this projected space yielded similar results, 93% performance in recognition and 98% in identification. We should comment that reducing the number of features using PCA increases the robustness, due to estimating less parameters from a limited training sample and the testing part is computationally less expensive, however some loss in the performance is also observed.

Some of our experimental results are summarized in Table 1. Numbers in the parenthesis show the number of features used. The first column in the table shows the results of using 12 geometric features. Second column gives the results obtained using Keren's algebraic invariants calculated for four fingers, excluding the thumb. Third column shows the results obtained by combining the features of both methods. As it is clearly seen, while the fusion of the methods increased the identification success above to 95%, the verification rate increased above to 99% and the false acceptance rate decreased down to 1%. The last column in Table 1 illustrates results observed after the PCA has been performed.

	Geom (12)	*IP (12)*	*Geom.+ IP (16)*	*PCA (5)*
Ident.	**88%**	**84%**	**95%**	**93%**
Verif.	**97%**	**90%**	**99%**	**98%**

Table 1. Shows the results obtained using different methods.

5. Conclusions

In this paper, implicit polynomials have been proposed for recognizing hand shapes and the results are compared with existing methods while researching the best features for identification-verification tasks. The results show that the fusion of invariants from the implicit polynomials and geometric features improve the performance of identification and verification. Future work may include using dynamic training data; and observing the long run performance of the recognition system, which has the ability to update its training data during each use.

Combining multiple classifiers could also increase system's reliability. A more successful hand recognition system will contribute to the other biometric methods' effectiveness and can widely be used in applications that require low-medium security.

References

[1] "HaSIS -A Hand Shape Identification System", www.csr.unibo.it/research/ Biolab/*hand.htm*

[2] Civi, H., "Implicit Algebraic Curves and Surfaces for Shape Modelling and Recognition", Ph.D. Thesis, Bogazici University, 1997.

[3] Jain, A.K., Ross, A. and Pankanti, S., "A Prototype Hand Geometry-Based Verification System", Proceedings of Second International Conference on Audio- and Video-based Biometric Person Authentication, Washington D.C., USA, pp. 166-171, 1999.

[4] Keren, D., "Using Symbolic Computation to Find Algebraic Invariants", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, No. 11, pp. 1143-1149, November 1994.

[5] Keren, D., D. Cooper, and J. Subrahmonia, "Describing Complicated Objects by Implicit Polynomials", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, No. 1, pp. 38-53, January 1994.

[6] Lei Z., Blane M. M. and Cooper D. B. "3L Fitting of Higher Degree Implicit Polynomials", Proceedings of Third IEEE Workshop on Applications of Computer Vision, Saratosa, FL, December 1996.

[7] Miller, B., "Vital Signs of Identity", IEEE Spectrum, Vol. 31, No. 2, pp. 22-30, February 1994.

[8] Sanchez-Reillo, R., Sanchez-Avila and C., Gonzales-Marcos A., "Biometric Identification Through Hand Geometry Measurements", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 10, pp. 1168-1171, October 2000.

[9] Tasdizen, T., Tarel, J.P. and Cooper, D.B., "Improving the Stability of Algebraic Curves for Applications", IEEE Transactions on Image Processing, Vol. 9, No. 3, pp. 405-416, March 2000.

[10] Taubin, G., "Estimation of Planar Curves, Surfaces and Nonplanar Space Curves defined by Implicit Equations, with Applications to Edge and Range Image Segmentation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 13, No. 11, pp 1115-1138, November 1991.

[11] Unsalan, C., Ercil, A. "A New Robust and Fast Technique for Implicit Polynomial Fitting", Proceedings of M^2VIP, pp. 15-20, September 1999.

[12] Wolovich, W. A., M. Unel, The Determination of Implicit Polynomial Canonical Curves Using Quartic Factorizations, Brown University, LEMS Laboratory Technical Report, LEMS-162, March 1997.

[13] Wolovich, W. A., M. Unel "Algebraic invariants for implicit polynomial curves," Brown University, LEMS Laboratory Technical Report, LEMS-165, May 1997.

[14] Yalniz, M., "Algebraic Invariants for Implicit Polynomials Case Study: Patterns in Turkish Hand-woven Carpets", M.S. Thesis, Bogazici University, 1999.