Multilevel Algebraic Invariants Extraction by Incremental Fitting Scheme

Bo Zheng[1], Jun Takamatsu[2], and Katsushi Ikeuchi[1]

[1] Institute of Industrial Science, The University of Tokyo, Komaba 4-6-1, Meguro-ku, Tokyo, 153-8505 JAPAN. E-mail: {zheng,ki}@cvl.iis.u-tokyo.ac.jp
[2] Nara Institute of Science and Technology (NAIST), 8916-5 Takayama, Ikoma, NARA 630-0192 JAPAN. E-mail: j-taka@is.naist.jp

Abstract. Algebraic invariants extracted from coefficients of implicit polynomials (IPs) have been attractive because of its convenience for solving the recognition problem in computer vision and image analysis. However, traditional IP fitting methods fixed the polynomial degree and thus lead to the difficulty for obtaining specific invariants according to the complexity of an object. In this paper, we propose a multilevel method for invariant extraction based on an incremental fitting scheme. Because this fitting scheme incrementally determines the coefficient set of IP, we can extract the invariants from different degree forms of IP coefficients during the incremental procedure. Our method is effective, not only because it adaptively encodes the invariants according to the complexity of shapes, but also we encodes the information of evaluating the contribution of representation to each degree invariant set, so as to have better discriminability. Experimental results demonstrate the effectiveness of our method compared with prior methods.

1 Introduction

The feature of shapes encoded by *Implicit Polynomials* (IPs) plays an essential role in computer vision for solving the problems on recognition [1–7], registration [8, 2, 9–11] and matching [3, 2, 12, 10], because it is superior in the areas of fast fitting, algebraic/geometric invariants, few parameters and robustness against noise and occlusion. There have been great improvements concerning IPs with its increased use during the late 1980s and early 1990s [3, 1]; Recently, new robust and consistent fitting methods like 3L fitting [13], gradient-one fitting [14], Rigid regression [14, 15], and incremental fitting [16] make them feasible for real-time applications for object recognition tasks.

The main advantage of implicit polynomials for recognition is the existence of algebraic/geometric invariants, which are functions of the polynomial coefficients that do not change after a coordinate transformation. The algebraic invariants that are found by Taubin and Cooper [8], Teral and Cooper [2], Keren [1] and Unsalan [17] 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 [7].

However, these invariant extraction methods are based on the traditional fitting scheme that usually fixed the degree of IP for in fitting procedure, regardless of the complexity of shapes. From the view of accurate shape description, there is no doubt that a complex object requires more invariants encoded by higher degree IP, whereas a simple object only needs less invariants encoded by lower degree IP. Therefore, unfortunately the method of using invariants encoded by same degree IPs leads to the difficulty on recognition accuracy, especially while dealing with a large database holding various shapes.

Our previous work proposed an incremental scheme for IP fitting that can adaptively determine the degree of IP to be suitable for various shapes [16]. The method makes it possible that the invariants are extracted from IPs of different degrees according to the different complexity levels of shapes and obviously that would enhance the accuracy for recognition.

In this paper, we first extend our incremental scheme to make it suitable for invariants extraction. Then we propose multilevel invariant extraction method based on combination of Taubin's invariants, degree information and fitting accuracy information. The reported experimental results demonstrate the effectiveness of our method that can extract the richer invariants exactly compared with prior methods.

The rest of this paper is organized as follows. Section 2 reviews recent advances in IP modeling and introduces the incremental framework [16]. Section 3 presents our multilevel extraction method based on a form-by-form incremental scheme and practical method of combining the degree and fitting accuracy information. Section 4 reports our experimental results followed by conclusion in Section 5.

2 Incremental Scheme of Implicit Polynomial Fitting

2.1 Implicit Polynomial

IP is an implicit function defined in a multivariate polynomial form. For example, the 3D IP of degree n is denoted by:

$$f_n(\mathbf{x}) = \sum_{0 \le i,j,k; i+j+k \le n} a_{ijk} x^i y^j z^k = \sum_l a_l m_l(\mathbf{x}). \tag{1}$$

where $\mathbf{x} = (x\ y\ z)$ is a 3D data point; $m_l(\mathbf{x}) = x^i y^j z^k$ is the monomial function with the accompanying coefficient a_l. Note, the relationship between indices l and $\{i,j,k\}$ are determined by the inverse *lexicographical order* (see Tab. 1 in [16]).

The *homogeneous binary polynomial* of degree r, $h_r = \sum_{i+j+k=r} a_{ijk} x^i y^j z^k$, is called the r-th degree form of the IP, and the highest (n-th) degree form is called *leading form.* If h_r can be described as $h_r = \mathbf{p}_r^T \mathbf{m}_r$, where $\mathbf{p}_r$ is coefficient vector of the r-th form as: $\mathbf{p}_r = (a_{r,0,0}, a_{r-1,1,0}, \dots, a_{0,0,r})^T$ and $\mathbf{m}_r$ is the corresponding monomial vector: $\mathbf{m}_r = (x^r, x^{r-1}y, \dots, z^r)^T$, then the polynomial f_n

can be described as:

$$f_n = \sum_{r=0}^{n} h_r = \sum_{r=0}^{n} \mathbf{p}_r^T \mathbf{m}_r. \tag{2}$$

2.2 Least-squares Fitting

In general, building an IP model can be regarded as a regression problem, which approximates a multivariate IP function from scattered data. This typically formulates the issue as to solve the following minimization problem:

$$\min_{f} \sum_{i=1}^{N} dist(b_i, f(\mathbf{x}_i)), \tag{3}$$

where $dist(\cdot, \cdot)$ is a distance function; the common choice is the L_2 norm as: $dist(b_i, f(\mathbf{x})) = (b_i - f(\mathbf{x}_i))^2$. b_i is the offset term for implicit polynomial fitting which can be determined by different optimization constraints from a method such as 3L method [13].

Substituting the representation of Eq. (1) into Eq. (3), we get to minimize the convex differentiable objective function $E(\mathbf{a})$ as:

$$\min E(\mathbf{a}) = \min_{\mathbf{a} \in \mathcal{R}^m} (M\mathbf{a} - \mathbf{b})^2 = \min_{\mathbf{a} \in \mathcal{R}^m} \mathbf{a}^T M^T M\mathbf{a} - 2\mathbf{a}^T M^T \mathbf{b} + \mathbf{b}^T \mathbf{b}. \tag{4}$$

where M is the matrix whose l-th column $\mathbf{m}_l$ is $(m_l(\mathbf{x}_1), m_l(\mathbf{x}_2), \dots, m_l(\mathbf{x}_N))^T$ (see Eq. (1)); $\mathbf{a}$ is the unknown coefficient vector of IP; and $\mathbf{b}$ is the offset term vector.

The derivatives of $E(\mathbf{a})$ with respect to the variable $\mathbf{a}$ vanish for optimality:

$$\frac{\partial E}{\partial \mathbf{a}} = 2M^T M\mathbf{a} - 2M^T \mathbf{b} = 0, \tag{5}$$

which leads to the following linear system of equations to be solved.

$$M^T M\mathbf{a} = M^T \mathbf{b}. \tag{6}$$

Because solving the linear system (6) requires that the fixed IP degree must be assigned to construct the inverse of the coefficient matrix $M^T M$, none of the prior methods allow changing the degree during the fitting procedure.

2.3 Column-by-column Incremental Fitting Process

For designing an incremental scheme [16], Eq. (6) is solved by the QR decomposition method. That is, matrix M ($\in \mathcal{R}^{N \times m}$) is decomposed as: $M = QR$ ($Q \in \mathcal{R}^{N \times m}$ and $R \in \mathcal{R}^{m \times m}$), where Q satisfies: $Q^T Q = I$ (I is an $m \times m$ identity matrix), and R is an upper triangular matrix. Then, substituting $M = QR$

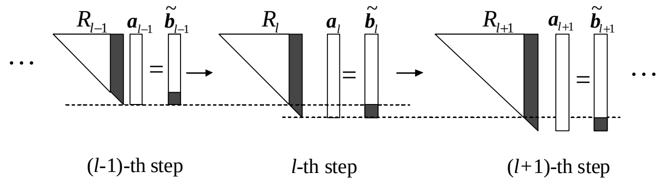

Fig. 1. The incremental scheme: dimension of the upper triangular linear system grows at each step and only the elements shown in dark gray need to be calculated.

into Eq. (6), we obtain a linear system with an upper-triangular coefficient matrix R derived as:

$$R^T Q^T Q R\mathbf{a} = R^T Q^T \mathbf{b} \rightarrow R\mathbf{a} = Q^T \mathbf{b} \rightarrow R\mathbf{a} = \widetilde{\mathbf{b}}. \tag{7}$$

In order to adapt the fitting to automatically determine the coefficients according to shape complexity, our previous work proposed a column-by-column incremental scheme by taking QR decomposition onto each dimensionally incremental matrix M, and then the upper-triangular linear system (7) will be solved incrementally until the desired fitting accuracy can be satisfied. This process can be generated by Gram-Schmidt process by continuously orthogonalizing the incoming column vector $\mathbf{m}_l$. This process is illustrated in Fig. 1, where the dimension of the upper triangular linear system increases, and thus the coefficient vector $\mathbf{a}$ with an incremental dimension can be obtained at any step by solving the triangular linear equation system in little computational cost.

This incremental scheme works efficiently because the calculation for dimension increment between two successive steps is computationally efficient. Fig. 1 illustrates this fact that, in this incremental scheme, only the right-most column of R_l and the last element of $\widetilde{\mathbf{b}}_l$ need to be calculated which are shown with dark gray blocks.

3 Multilevel Invariants Extraction

3.1 Form-by-form Incremental Scheme

The column-by-column Incremental method cannot be directly used for extracting the invariants from IP's leading form, because it cannot guarantee to obtain a complete leading form of IP which is necessary for calculating the invariants [8]. To make the incremental scheme suitable for invariants extraction, we first modify the incremental scheme to be in a form-by-form incremental manner; that is the monomial matrix will increase multiple columns corresponding to one form rather than increasing one column at each step. Thus, the incremental process is the iteration using the index r in Eq. (2) but not l in Eq. (1).

Fig. 2 illustrates this process, where the necessary elements for calculation at each step are shown in dark gray. Note, this process can be maintained also by

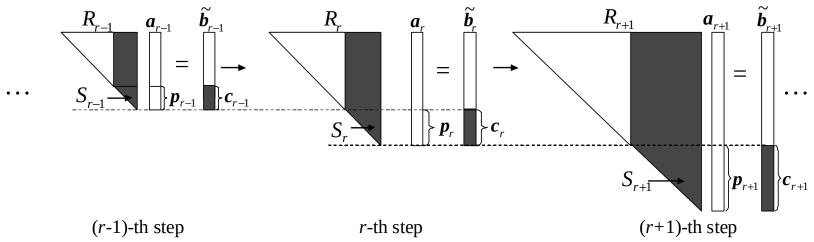

Fig. 2. The from-by-form incremental scheme: dimension of the upper triangular linear system grows by one form at each step and the elements shown in dark gray need to be calculated.

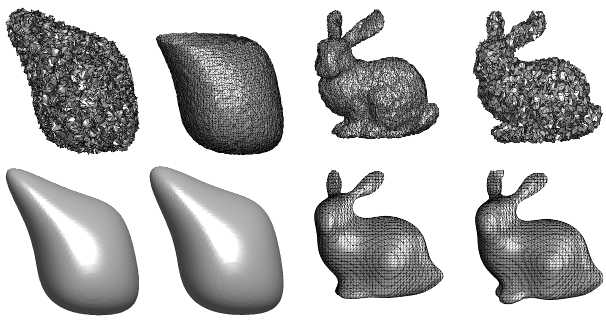

Fig. 3. Top: original objects in different noise levels or with missing data; Bottom: IP fits of degree 4, 4, 8 and 8 respectively.

the Gram-Schmidt process [16], the only difference is that we stop the iteration at the step when an integral form of IP can be obtained. The reason why we need do this is that the form-by-form incremental scheme guarantees that the fitting can be in the Euclidean invariant way [14] and thus we can extract the Euclidean invariants from the obtained forms of IP as described in following section.

Fig. 3 shows some fitting examples of our method. We can see that incremental fitting scheme can not only determine the degree for different shapes, but also keep the characteristic of IP, the robustness against noise and missing data.

3.2 Extracting Invariants by Incremental Scheme

A simple but practical method proposed by Taubin and Cooper [8] shows us that invariants of an IP can be extracted from a specific coefficient form, which is introduced briefly in Appendix A. Taubin and Cooper also showed that invariants obtained from the leading form are practically powerful in their recognition examples in [8]. Combining their method, our fitting method can be modified as a multilevel method for invariants extraction.

Fig. 2 also shows the fact that, at each step, we can obtain a triangular linear sub-system of equations:

$$S_r \mathbf{p}_r = \mathbf{c}_r, \tag{8}$$

where S_r is the right-lower triangular sub-matrix of R_r, and $\mathbf{c}_r$ is the lower sub-vector of $\widetilde{\mathbf{b}}_r$; then by solving this linear sub-system we can obtain the coefficient vector $\mathbf{p}_r$ corresponding to the r-th form of IP. Note, since $\mathbf{p}_r$ is always the highest degree form of the IP at current step, we call each $\mathbf{p}_r$ the leading form. Thus, we can design our descriptors to be a set of leading forms' invariants in various degrees.

We extract the invariants $\mathbf{I}_r$ from $\mathbf{p}_r$, the r-th leading form, by using the function of Eq. (16) in Appendix. If we arrange them into a vector $\mathbf{I}$ as: $\mathbf{I} = \{\mathbf{I}_2, \mathbf{I}_3, \ldots, \mathbf{I}_n\}$, then we take $\mathbf{I}$ as our shape descriptor. Note, for different objects descriptors of $\mathbf{I}$ do not have to keep in the same dimension, *e.g.*, two descriptors can be compared by only using the corresponding elements.

3.3 Weighting Function

For further enhancing discriminability, let us present a weighting method that can determine a weight w_r for the set of invariants $\mathbf{I}_r$ to evaluate how it is important for describe the shape. Then the invariant $\mathbf{I}$ can be modified as:

$$\mathbf{I} = \{w_2\mathbf{I}_2, w_3\mathbf{I}_3, \ldots, w_n\mathbf{I}_n\}. \tag{9}$$

The basic but effective evaluation for similarity of shape representation is the least-squares error E in Eq. (4). Fortunately, we find a fact that the incremental method can calculate the least-squares errors at each step in a simpler way. It only requires the computation related to right-hand vector $\widetilde{\mathbf{b}}$, *e.g.*, at the r-th step it can calculated as:

$$\begin{aligned}
E(\mathbf{a}_r) &= \mathbf{a}_r^T M_r^T M_r \mathbf{a}_r - 2\mathbf{a}_r^T M_r^T \mathbf{b} + \mathbf{b}^T \mathbf{b} \\
&= \mathbf{a}_r^T R_r^T Q_r^T Q_r R_r \mathbf{a}_r - 2\mathbf{a}_r^T R_r^T Q_r^T \mathbf{b} + \mathbf{b}^T \mathbf{b} \\
&= (R_r \mathbf{a}_r)^T (R_r \mathbf{a}_r) - 2(R_r \mathbf{a}_r)^T (Q_r^T \mathbf{b}) + \mathbf{b}^T \mathbf{b} \\
&= \widetilde{\mathbf{b}}_r^T \widetilde{\mathbf{b}}_r - 2\widetilde{\mathbf{b}}_r^T \widetilde{\mathbf{b}}_r + \mathbf{b}^T \mathbf{b} \\
&= - \parallel \widetilde{\mathbf{b}}_r \parallel^2 + \parallel \mathbf{b} \parallel^2,
\end{aligned} \tag{10}$$

where M_r is the matrix holding the monomial columns, and R_r, $\mathbf{a}_r$ and $\widetilde{\mathbf{b}}_r$ are respectively the upper triangular matrix, parameter vector, and right-hand vector at the r-step, see Fig. 2. From Eq. (10), we can see the fact that the closer $\parallel \widetilde{\mathbf{b}}_r \parallel^2$ to $\parallel \mathbf{b} \parallel^2$ (note $\mathbf{b}$ has been given and fixed in the procedure and $\parallel \widetilde{\mathbf{b}}_r \parallel^2 < \parallel \mathbf{b} \parallel^2$ always), the smaller is the least-squares error E.

Because $\parallel \widetilde{\mathbf{b}}_r \parallel^2$ can be represented as: $\parallel \widetilde{\mathbf{b}}_r \parallel^2 = \parallel \widetilde{\mathbf{c}}_1 \parallel^2 + \parallel \widetilde{\mathbf{c}}_2 \parallel^2 + \cdots + \parallel \widetilde{\mathbf{c}}_r \parallel^2$ and $\widetilde{\mathbf{c}}_k (k < r)$ are determined at the previous steps, we can say that the

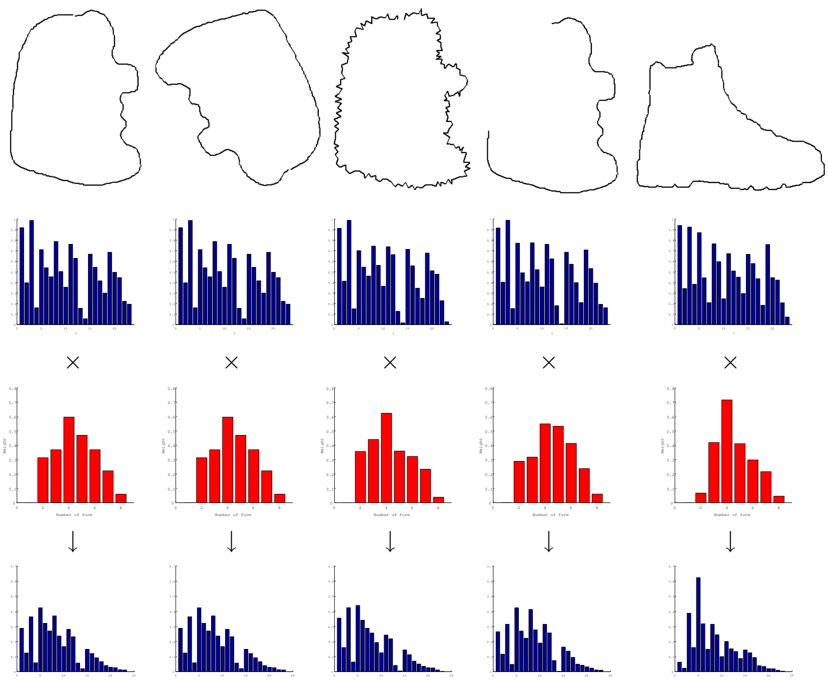

Fig. 4. Top row: original 2D objects; Second row: extracted invariants of each objects by degree increasing form 2 to 8; Third row: weights calculated by Eq. (11) according to step number; Bottom row: weighted invariants by Eq. (9).

larger $\parallel \widetilde{\mathbf{c}}_r \parallel$ is, the smaller the least-squares error is, and thus the current step has more contribution to the whole fitting process. Therefore the corresponding invariant vector $\mathbf{I}_r$ can be weighted whether it is important to the whole shape description, if we let the weight w_r in Eq. (9) be

$$w_r = \parallel \mathbf{c}_r \parallel^2 / \parallel \widetilde{\mathbf{b}} \parallel^2, \tag{11}$$

where we normalize the weights by right-hand vector $\widetilde{\mathbf{b}}$ which is obtained at the final step. Fig. (4) illustrates the above process by encoding the invariants to 5 samples.

3.4 Dissimilarity Evaluation

For practical object recognition, three factors are considered for accurate shape discrimination: weighting invariants, determined fitting degree and final fitting accuracy. Thus, a more robust dissimilarity evaluation for discriminating two objects $\mathcal{O}_1$ and $\mathcal{O}_2$ can be described as:

$$Dis(\mathcal{O}_1, \mathcal{O}_2) = \alpha \parallel \mathbf{I1} - \mathbf{I2} \parallel + \beta |D_1 - D_2| + \gamma \parallel E_1 - E_2 \parallel, \tag{12}$$

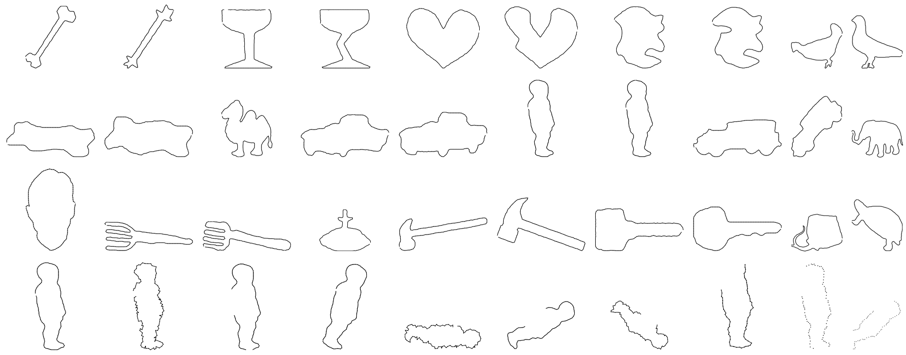

Fig. 5. Top 3 rows: original samples in LEMS216. Bottom row: randomly generated samples by adding noise and missing data and operated by Euclidean transformation

Table 1. The recognition rate comparison for the fixed fitting method and our method.

	degree	4	6	8	10
3L method	recognition rate	82.1%	87.7%	86.4%	81.3%
RR method	recognition rate	81.1%	91.0%	92.1%	85.3%
	parameters: α	1	1	1	1
	β	0	0.1	0	0.1
	γ	0	0	0.2	0.2
our method	recognition rate	92.3%	95.1%	94.3%	97.8%

where **I1** and **I2** are invariant vectors defined in Eq. (9); D_1 and D_2 are obtained fitting degrees; and E_1 and E_2 are the final fitting errors defined by Zheng *et al.* in [16] respectively for two objects. Also they are controlled by three weighting parameters: α, β and γ.

4 Experimental Result

4.1 Experiment on Object Recognition

In this experiment on object recognition, we adopted 2D shape database LEMS216, consisting of 216 samples divided into 16 categories [18]. But the database was too small for examining effectiveness, and we extended LEMS216 to a larger database of 2160 samples by adding 9 variations for each of original samples in LEMS216, through adding noises, missing data, and random Euclidean transformation. Some selected original and generated samples are shown in Fig. 5. Note, all the shape data sets are regularized by centering the data-set center of mass at the origin of the coordinate system and scaling it by dividing each point by the average length from points to origin.

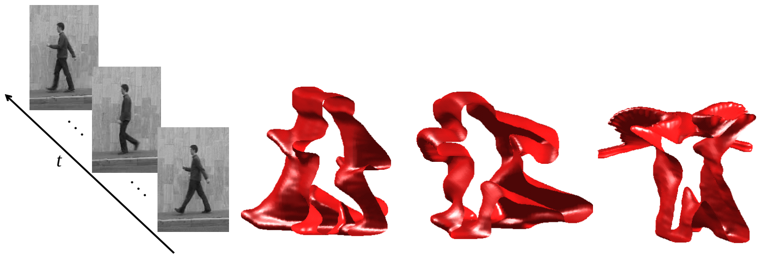

Fig. 6. Left: video frames for silhouettes extraction. the others: extracted 3D models of "walking","running" and "jumping" from three videos.

We generated another 100 samples with a similar operation for test data set, and then we searched for their matches in the database. We tested three methods: two degree-fixed fitting methods: 3L method [13] and Rigid Regression (RR) method [14] and our method, and for both degree-fixed methods same parameters were employed for 3L method ($\varepsilon = 0.05$) and RR method ($\kappa = 10^{-4}$). For the degree-fixed fitting method, the invariants were extracted from the leading form of a degree-fixed IP by using Taubin and Cooper's method described in Appendix. In Tab. 1, we show comparable results on recognition rate in different cases by changing degree for fixed methods and setting different parameters for Eq. 12 for our method.

4.2 Experiment on Action Recognition

M. Blank *et al.* [19] proposed a method for human action recognition that models human actions from video data. To this application, we use IP model the action defined by continuous silhouettes extracted from video frames (see Fig. 6). IP has the capability to model global features of objects that can make the action recognition feasible for fast implementation. We detected three actions, "walking","running" and "jumping", from ten videos provided by M. Blank *et al.* [19]. Our fast invariants extraction method brought better performance that the overall processing time (incrementally modeling and extracting invariants) for each detection can be completed within about 2.5 seconds, compared to the result (30 seconds) in [19].

5 Conclusions

In this paper, we represent a method that can efficiently extract algebraic invariants from the shape-representing IPs. The quality of the extracted invariants is first demonstrated by the success of the incremental scheme that can extract invariants according to the complexity of shapes. Furthermore the performance is enhanced by combining weighting function of representing contribution and the information of degrees and fitting accuracy. The reported experimental results showed that it has capability to cope with various 2D/3D recognition tasks and has a potential for real-time applications.

Appendix

A. Taubin and Cooper's Invariants

Let us briefly describe how to extract invariants vector $\mathbf{I}_r$ from the r-th form vector $\mathbf{p}_r$ once it has been solved out by Eq. (8). Taubin and Cooper [8] proposed a simple symbolic computation to achieve this. It can be described as follows.

Let coefficient a_{ijk} in $\mathbf{p}_r$ (also see Eq. (1)) be presented as $\frac{\Phi_{ijk}}{i!j!k!}$ and the coefficient vector of the r-th form of a polynomial be presented as $\mathbf{\Phi}_{[r]} = (\frac{\Phi_{r00}}{\sqrt{r!0!0!}}\ \frac{\Phi_{r-1,1,0}}{\sqrt{(r-1)!1!0!}}\ \cdots\ \frac{\Phi_{00r}}{\sqrt{0!0!r!}})^T$. Then there exists a matrix $\mathbf{\Phi}_{[s,t]}$ whose singular values are invariant under Euclidean transformation, according to a symbolic computational manner:

$$\mathbf{\Phi}_{[s,t]} = \mathbf{\Phi}_{[s]} \star \mathbf{\Phi}_{[t]}^T, \tag{13}$$

where $\star$ represents the classic matrix multiplication with the difference that the individual element-wise multiplications are performed according to the rule $\frac{\Phi_{ijk}}{\sqrt{i!j!k!}} \star \frac{\Phi_{abc}}{\sqrt{a!b!c!}} = \frac{\Phi_{i+a,j+b,k+c}}{\sqrt{i!j!k!a!b!c!}}$. For example, if

$$\mathbf{\Phi}_{[1]} = (\frac{\Phi_{100}}{\sqrt{1!0!0!}}\ \frac{\Phi_{010}}{\sqrt{0!1!0!}}\ \frac{\Phi_{001}}{\sqrt{0!0!1!}})^T, \tag{14}$$

then

$$\begin{aligned}\mathbf{\Phi}_{[1,1]} = \mathbf{\Phi}_{[1]} \star \mathbf{\Phi}_{[1]}^T &= \begin{pmatrix} \Phi_{200} & \Phi_{110} & \Phi_{101} \\ \Phi_{110} & \Phi_{020} & \Phi_{011} \\ \Phi_{101} & \Phi_{011} & \Phi_{002} \end{pmatrix} \\ &= \begin{pmatrix} 2a_{200} & a_{110} & a_{101} \\ a_{110} & 2a_{020} & a_{011} \\ a_{101} & a_{011} & 2a_{002} \end{pmatrix}. \end{aligned} \tag{15}$$

Then the eigenvalues (singular values for non-square matrix) of $\mathbf{\Phi}_{[1,1]}$ are invariant under Euclidean transformation. Also we can see that matrix $\mathbf{\Phi}_{[1,1]}$ can be constructed by only using the coefficients in $\mathbf{p}_2 = \{a_{200}, a_{110}, a_{101}, a_{020}, a_{011}, a_{002}\}$ of the 2nd form. Therefore, as a result, if the r-th form of coefficients $\mathbf{p}_r$ is given, we can construct a matrix $\mathbf{\Phi}_{[s,t]}$ $(s+t=r)$ and calculate out the singular values by SVD worked as the Euclidean invariants. Furthermore, the normalized singular values are scaling invariant. For more details and proofs, let us refer to [8]. Note, in this paper, as a matter of convenience, we only employ the invariants from r-th degree form which are eigenvalues of $\mathbf{\Phi}_{[r/2,r/2]}$ if r is even, or singular values of $\mathbf{\Phi}_{[(r-1)/2,(r+1)/2]}$ otherwise. Let us denote this symbolic computation by a function $Inv(\cdot)$:

$$\mathbf{I}_r = Inv(\mathbf{p}_r), \tag{16}$$

where $\mathbf{I}_r$ is the returned invariants holding the singular values of $\mathbf{\Phi}$.

References

1. Keren, D.: Using symbolic computation to find algebraic invariants. IEEE Trans. on Patt. Anal. Mach. Intell. **16**(11) (1994) 1143–1149
2. Tarel, J., Cooper, D.: The Complex Representation of Algebraic Curves and Its Simple Exploitation for Pose Estimation and Invariant Recognition. IEEE Trans. on Patt. Anal. Mach. Intell. **22**(7) (2000) 663–674
3. Taubin, G.: Estimation of Planar Curves, Surfaces and Nonplanar Space Curves Defined by Implicit Equations with Applications to Edge and Range Image Segmentation. IEEE Trans. on Patt. Anal. Mach. Intell. **13**(11) (1991) 1115–1138
4. Subrahmonia, J., Cooper, D., Keren, D.: Practical reliable bayesian recognition of 2D and 3D objects using implicit polynomials and algebraic invariants. IEEE Trans. on Patt. Anal. Mach. Intell. **18**(5) (1996) 505–519
5. Oden, C., Ercil, A., Buke, B.: Combining implicit polynomials and geometric features for hand recognition. Pattern Recognition Letters **24**(13) (2003) 2145–2152
6. Marola, G.: A Technique for Finding the Symmetry Axes of Implicit Polynomial Curves under Perspective Projection. IEEE Trans. on Patt. Anal. Mach. Intell. **27**(3) (2005)
7. Wolovich, W.A., Unel, M.: The Determination of Implicit Polynomial Canonical Curves. IEEE Trans. on Patt. Anal. Mach. Intell. **20**(10) (1998) 1080–1090
8. Taubin, G., Cooper, D.: Symbolic and Numerical Computation for Artificial Intelligence, chapter 6. Computational Mathematics and Applications. Academic Press (1992)
9. Forsyth, D., Mundy, J., Zisserman, A., Coelho, C., Heller, A., Rothwell, C.: Invariant Descriptors for 3D Object Recognition and Pose. IEEE Trans. on Patt. Anal. Mach. Intell. **13**(10) (1991) 971–992
10. Zheng, B., Takamatsu, J., Ikeuchi, K.: 3D Model Segmentation and Representation with Implicit Polynomials. To appear in IEICE Trans. on Information and Systems **E91-D**(4) (2008)
11. Zheng, B., Ishikawa, R., Oishi, T., Takamatsu, J., Ikeuchi, K.: 6-DOF Pose Estimation from Single Ultrasound Image using 3D IP Models. Proc. IEEE Conf. Computer Vision and Patt. Rec. (CVPR) Workshop on Ojb. Trac. Class. Beyond Visi. Spec. (OTCBVS'08) (2008)
12. Tarel, J.P., Civi, H., Cooper, D.B.: Pose Estimation of Free-Form 3D Objects without Point Matching using Algebraic Surface Models. In Proceedings of IEEE Workshop Model Based 3D Image Analysis (1998) 13–21
13. Blane, M., Lei, Z.B., Cooper, D.B.: The 3L Algorithm for Fitting Implicit Polynomial Curves and Surfaces to Data. IEEE Trans. on Patt. Anal. Mach. Intell. **22**(3) (2000) 298–313
14. Tasdizen, T., Tarel, J.P., Cooper, D.B.: Improving the Stability of Algebraic Curves for Applications. IEEE Trans. on Imag. Proc. **9**(3) (2000) 405–416
15. Sahin, T., Unel, M.: Fitting Globally Stabilized Algebraic Surfaces to Range Data. Proc. IEEE Conf. Int. Conf. on Comp. Visi. **2** (2005) 1083–1088
16. Zheng, B., Takamatsu, J., Ikeuchi, K.: Adaptively Determining Degrees of Implicit Polynomial Curves and Surfaces. **LNCS4844** (2007) 289–300
17. Unsalan, C.: A Model Based Approach for Pose Estimation and Rotation Invariant Object Matching. Pattern Recogn. Lett. **28**(1) (2007) 49–57
18. : http://www.lems.brown.edu/vision/software/index.html
19. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: Proc. IEEE Conf. Int. Conf. on Comp. Visi. (2005) 1395–1402