Local Representation of 3D Free-Form Contours for Pose Estimation

Marco A. Chavarria, Gerald Sommer
Cognitive Systems Group
Institute of Computer Sciences and Applied Mathematics
Chistian-Albrechts-University of Kiel, 24118 Kiel, Germany
{mc,gs}@ks.informatik.uni-kiel.de

Abstract

In this paper we present a new representation for 3D free-form contours in the conformal geometric algebra $\mathcal{G}_{4,1}$. This new representation allows to extract local geometrical feature information which is used to solve the correspondence problem for pose estimation applications. Under perspective projection, local features are extracted from a projected contour segment and compared with image features obtained from the monogenic signal. We tested our approach using synthetical and real data for several pose estimation algorithms.

1. Introduction

Commonly a 3D free-form contour is represented by a set of chained points or by a set of parametrical functions, one function for every coordinate axis, see [8]. Rosenhahn and Sommer [6] use sets of coupled twist to model free-form contours and to gain spectral domain representations by applying the Fourier transform. This global representation is embedded in conformal geometric algebra, which allows to embed all the mathematical entities involved in the pose estimation problem in the same mathematical frame work.

In general, the main problem to solve before applying any pose estimation algorithm is the correspondence problem. Rosenhahn et al. [5] uses a modified ICP algorithm to find correspondences for occluded objects. In this case it is possible to select a threshold value to eliminate possible outliers. A disadvantage of this method is that it only works when the difference between the previous and actual pose is very small. To handle this problem in a more robust way, local features which characterize image and model points are needed.

To extract image features we use the monogenic signal approach presented by Felsberg and Sommer [2], which provides local amplitude, phase and orientation information. In a later work [3], an extension of the monogenic signal approach was presented to deal with phase based image processing problems like edge detection in a monogenic scale-space.

This paper is organized as follows: in section 2 we introduce the concepts of the entities in geometric conformal algebra which we use. A brief overview of the monogenic scale-space representation is presented in section 3. Instead of the global approach used in [5] and [7], we present in section 4 a local representation for 3D free-form contours, based on approximations of contour segments with motors (entities defined in conformal space as a combination or translations and rotations). This allows to get information in 3D space like local orientation and curvature. Since, the main idea is to fit 3D contour models to images, it is desirable to find model features which are compatible with image features. For this purpose, the contour segments and their features are projected onto the image plane to get this feature compatibility. We present in section 5 an ICP algorithm which combines local orientation information derived from our local model representation and from the image to get correspondences for the case of occluded objects in a more robust way. Finally, some experimental results are presented in section 6.

2. Entities in Conformal Geometric Algebra

An introduction to the basic concepts of conformal geometric algebra can be found in [6]. Points $\mathbf{X} \in \mathcal{G}_{3,1}$, lines $\mathbf{L} \in \mathcal{G}_{3,1}$ or planes $\mathbf{P} \in \mathcal{G}_{3,1}$ in projective geometric algebra can be embedded to the conformal geometric algebra $\mathcal{G}_{4,1}$ as shown in equation (1). This leads to dual representation of these entities.

$$\begin{array}{lllllll}\mathbf{e} \wedge \mathbf{X} & = & \underline{\mathbf{X}}^* & = & \mathbf{ex} + \mathbf{E} & \in & \mathcal{G}_{4,1} \\ \mathbf{e} \wedge \mathbf{L} & = & \underline{\mathbf{L}}^* & = & \mathbf{Er} + \mathbf{em} & \in & \mathcal{G}_{4,1} \\ \mathbf{e} \wedge \mathbf{P} & = & \underline{\mathbf{P}}^* & = & \mathbf{En} + \mathbf{e}d\mathbf{I_E} & \in & \mathcal{G}_{4,1},\end{array} \qquad (1)$$

A line is parameterized with the vectors $\mathbf{m}$ and $\mathbf{r}$, which stand for moment and direction respectively. A plane is parameterized by its normal vector $\mathbf{n}$ and the distance d to the origin. We use this dual representations in the next sections. To simplify the notation we will neglect the *-sign .

Motors in conformal geometric algebra $\mathbf{M} \in \mathcal{G}_{4,1}$ are expressed as a consecutive application of rotations and translations. Then, a motor can model a rotation of any entity around an arbitrary line in space. The idea is to translate the entity with the direction vector between the line and the origin, perform the rotation and translate it back to its original position. The exponential representation of a motor is shown in equation (2), where **m** and **r** are the parameters of the rotation line.

$$\begin{aligned} \mathbf{M} &= \exp\left(-\tfrac{\theta}{2}(\mathbf{r} + \mathbf{e}\mathbf{m})\right) \\ \mathbf{T} &= \exp\left(\tfrac{\mathbf{e}\mathbf{t}}{2}\right) \end{aligned} \tag{2}$$

A translator $\mathbf{T}$ is a special rotation acting at infinity, where $\mathbf{t} \in \mathcal{G}_3$ is the Euclidean translation vector.

3. Monogenic Signal

In [2] the monogenic signal of an image is defined as a combination of the original image and its Riesz transform to gain local geometric and structural information (amplitude, orientation and phase). An extension of this idea is the monogenic scale-space representation of an image [3]. If $p(\mathbf{x}; s)$ and $q(\mathbf{x}; s)$ are the filter responses of an image convolved with the Poisson and conjugate Poisson kernels respectively, local amplitude $A(\mathbf{x}; s)$ and phase $r(\mathbf{x}; s)$ are obtained for a scale s as shown in equation (3).

$$\begin{aligned} A(\mathbf{x}; s) &= \sqrt{|q(\mathbf{x}; s)|^2 + |p(\mathbf{x}; s)|^2} \\ r(\mathbf{x}; s) &= \tfrac{q(\mathbf{X}; s)}{|q(\mathbf{X}; s)|} \arctan\left(\tfrac{|q(\mathbf{X}; s)|}{p(\mathbf{X}; s)}\right). \end{aligned} \tag{3}$$

The local amplitude is related to the local energy of the signal. The local orientation and local phase are combined in the local phase vector. The local phase gives information about the local symmetry of the signal and the local orientation gives the direction of the highest signal variance. We use the differential phase congruency approach presented in [3] to detect edges form local amplitude and phase information. Then, for an edge point we get the local features orientation and phase angles in x and x directions.

$$F_i^{im} = \{\phi_i, \| r_i^x \|, \| r_i^y \|\}. \tag{4}$$

4. Local Free-form Contours

The aim of our proposed local representation is to approximate a contour point and its neighbors by a motor. This local approach is in contrast to the global approach used in [7], where motors have been used to model complete contours. The motor parameters (rotation line and angle) can be obtained geometrically from the parameters of a circle (radius and center). The figure 1 shows the basic idea of getting a circle from 3 points. The algebraic description of a circle in conformal geometric algebra is the dual of the outer product of these three points. Because three points, $\underline{\mathbf{X}}_{i-1}, \underline{\mathbf{X}}_i, \underline{\mathbf{X}}_{i+1} \in \mathcal{G}_{4,1}$ define a plane, parameterized with its normal $\mathbf{n}$ and distance d, the problem is locally mapped to that plane to make the computations easier.

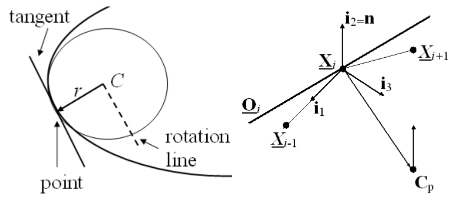

Figure 1. Left figure: local rotor for a 3D contour, right figure: local coordinate system needed to get the circle parameters and the motor.

In order to map the points (3D) in the plane (2D), a local coordinate system must be defined. To simplify the calculations we use the Euclidian part of the contour points, that is $\mathbf{x}_i, \mathbf{x}_{i-1}, \mathbf{x}_{i+1} \in \mathcal{G}_3$, and we set the origin in $\mathbf{x}_i$ as shown in the figure 1. The basis vectors are given by:

$$\mathbf{i}_1 = \tfrac{\mathbf{x}_i - \mathbf{x}_{i-1}}{\|\mathbf{x}_i - \mathbf{x}_{i-1}\|} \quad \mathbf{i}_2 = \mathbf{n} \quad \mathbf{i}_3 = \tfrac{\mathbf{i}_1 \underline{\times} \mathbf{i}_2}{\|\mathbf{i}_1 \underline{\times} \mathbf{i}_2\|}, \tag{5}$$

The operator $\underline{\times}$ is defined in the geometric algebra as the commutator product (in our case, its equivalent operation in Euclidian 3D space is the cross product). Once that the local coordinate system is defined, it is possible to get the coordinates of the points in the plane, defined by the basis vectors $\mathbf{i}_1$ and $\mathbf{i}_3$. As can be seen in the figure 1, the center of the circle $\mathbf{C}_p = a\mathbf{i}_1 + b\mathbf{i}_3$ and the radius vector $\mathbf{r_i}$ are easily calculated. The rotation line of the motor in 3D is obtained with the center of the circle and the normal vector $\mathbf{n}$ and the rotation angle θ_i is the angle defined by the segment $\overline{\underline{\mathbf{X}}_{i-1}\mathbf{C}_p\underline{\mathbf{X}}_{i+1}}$.

4.1. Contour Features

For every point of the 3D contour a local rotor is constructed, then several features can be extracted. We define the different features for a point $\underline{\mathbf{X}}_\mathbf{i}$ by

$$F_i^{3D} = \{\underline{\mathbf{O}}_i, \theta_i, \|\mathbf{r}_i\|\}, \tag{6}$$

where $\underline{\mathbf{O}}_i \in \mathcal{G}_{4,1}$ is the local orientation line. It represents the orientation of the contour segment in the 3D coordinate system as the tangent line at the point $\underline{\mathbf{X}}_i$. The rotation angle θ_i describes the amount of rotation needed to approximate the contour points and $\|\mathbf{r}_i\|$ is the radius of curvature.

In the cases where a local segment is described as a translator, from the direction of the translation the orientation line is calculated and the radius of curvature goes to infinity.

To find contour features, which are compatible with the image features (orientation and phase) obtained from the monogenic signal, we project the contour point and its orientation line onto the image plane. As can be seen in figure 2, from the projected line a normalized orientation vector $\mathbf{o}_i \in \mathcal{G}_{2,1}$ is calculated and the corresponding orientation angle α_i.

The concept of phase in the image plane delivers information of the local structure of the image. In the case of edges, the phase encodes a transition from one gray value to another in x and y directions. For 3D contours it is not possible to compute directly phase information in that sense, even if we project them onto the image plane. The monogenic signal can be calculated only from gray value images which contain structural information. An image with an artificially projected 3D contour lacks the structural information to compute the monogenic signal and therefore the phase. Despite of that, it is possible to assign a feature value for a contour point that represents such transition under certain assumptions. We call this feature transition index. The figure 2 shows an example of transitions in x and y directions for a point, under the assumption that our model represents a dark object on a bright background (or vice versa). The transition takes a value depending on the orientation of the vector $\mathbf{o}_i$ and the object assumption. Thus, for a projected contour point we obtain as a features the orientation and transition indexes

$$F_i^{con} = \{\alpha_i, t_i^x, t_i^y\}. \tag{7}$$

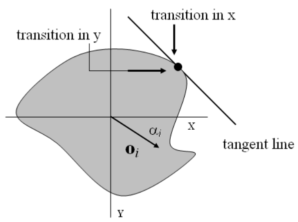

Figure 2. The transition index is determined according to the position of the normalized orientation vector in image coordinates and the object assumption (dark object on a bright background).

5. ICP with Local Features Approach

We combine an ICP algorithm with local features to find correspondences between segment pairs (image edges and contour segments) by analyzing its local structure. Since we make this analysis locally, there is no need of complete contour information to find correspondences.

The first step is to project a contour point and to find its nearest image edge point. Then, an edge following algorithm is applied to get edge segments of a determined length n. For each edge segment its orientation and phase values are stored in the feature vectors $\{\mathbf{a}_{im}, \mathbf{b}_{im}\}$ respectively. The next step is to find the nearest contour segment to the detected edge segment and to interpolate it to get the same number of points (contour and edge). Similarly, feature vectors are generated from contour segments containing the orientation and transition index $\{\mathbf{a}_{con}, \mathbf{b}_{con}\}$.

We use the feature correlation matrix to measure how strong our model and image features are related. For two feature vectors $\mathbf{a}_{con}$ and $\mathbf{a}_{im}$ it is given by

$$\mathbf{Corr}(\mathbf{a}_{con}, \mathbf{a}_{im}) = \frac{\mathbf{Cov}(\mathbf{a}_{con}, \mathbf{a}_{im})}{\sqrt{V_{con} V_{im}}}, \tag{8}$$

where $\mathbf{Cov}(\mathbf{a}_{con}, \mathbf{a}_{im})$ is the covariance matrix and V_{con}, V_{im} are the diagonal elements of the covariance matrix. The correlation may vary in a range of $-1 \leq \mathbf{Corr}(\mathbf{a}_{con}, \mathbf{a}_{im}) \leq \mathbf{1}$, where -1 indicates perfect negative correspondence , 0 indicates no correspondence and 1 indicates perfect correspondence. A segment pair is considered as outlier if its correlation is below some threshold value.

This process is repeated to get as many correspondence pairs as possible.

6. Experiments

We tested our correspondence search approach on artificial images for two different contour models. These are a mouse model which consists of curved segments and a cactus model which has mainly straight lines. Then the pose was calculated with these correspondences. Three pose estimation algorithms were tested: point-line [7], projective [1] and orthogonal iteration [4]. In figure 3 the average pose error is shown.

Occlusions on the models were simulated in a syntectic image. As can be seen in figure 3 for the mouse model the point-line and orthogonal iteration algorithms have an acceptable error at about 30% of occlusion and at about 20% in case of the cactus model. The mouse contains only curve segments and the correspondences can be found properly. For models containing more line segments like the cactus, the correspondence search algorithm are more sensitive to errors and therefore to occlusions. This results from the fact that a straight line has the same orientation for all its points, which causes uncertainty while searching correspondences. In both cases the projective pose estimation algorithm seems to be more robust with respect to occlusions and wrong correspondences.

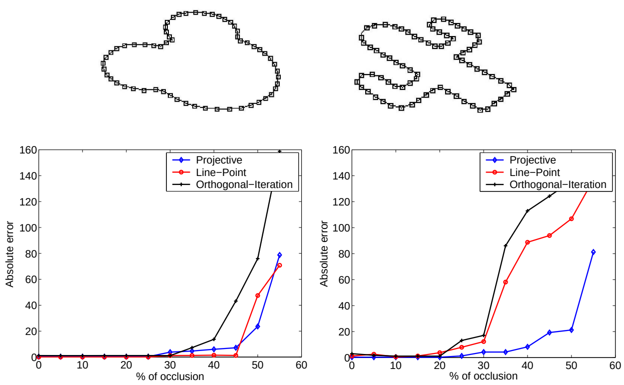

Figure 3. Absolute error in millimeters for the mouse model (left) and the cactus model (right) while increasing the occlusion.

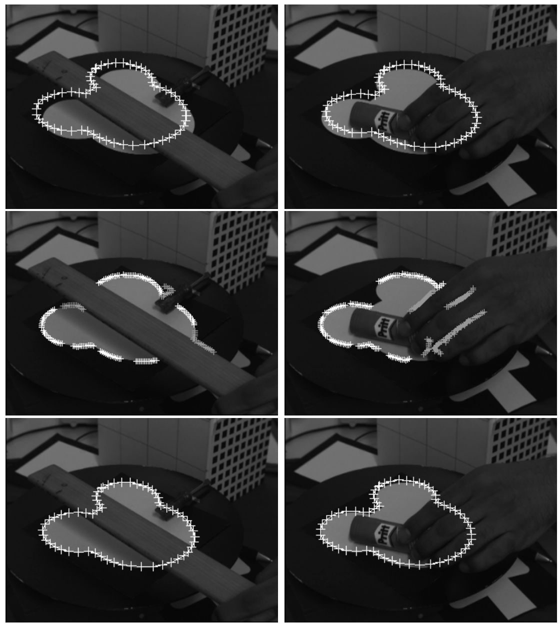

Figure 4. The top images show the initial pose, the middle row shows the detected segments and the bottom row shows the pose results.

Examples for experiments on real images are presented in the figure 4. The top row shows the initial position of the contour model, the next row shows all the detected contour segments. The detected segments which correspond to occlusions are removed as outliers. This happens when we have no correlation or negative correlation between image features (orientation and phase) and model features (orientation and transition index). The bottom row shows the fitting of the model with the image contour.

7. Conclusions and Future Work

In this paper we presented a new representation for 3D free-form contours, which delivers local geometrical information in 3D space and in the image plane. The local information allows us to couple model features and image features to solve the correspondence problem when the object is partially occluded. Since we are using the local orientation as a feature, the correspondence search works better for models containing mainly curved segments.

The representation in the conformal geometric algebra allows to treat a single point or a segment as an element of the algebra, making our local representation compatible with the pose estimation constraints presented by Rosenhahn and Sommer [7]. An extension of this approach can also be applied for modelling 3D surfaces and extracting 3D silhouettes from surfaces. Since the idea of this local approach is similar to the kinematic chains, it could also be applied to the problem of pose estimation of contours and surfaces with local deformations.

References

[1] H. Araujo, R. Carceroni, and C. Brown. A fully projective formulation to improve the accuracy of lowe's pose-estimation algorithm. 70(2):227–238, May 1998.

[2] M. Felsberg and G. Sommer. The monogenic signal. *IEEE Transactions on Signal Processing*, 49(12):3136–3144, December 2001.

[3] M. Felsberg and G. Sommer. The monogenic scale-space: A unifying approach to phase-based image processing in scale-space. *J. Math. Imaging Vis.*, 21(1):5–26, 2004.

[4] C.-P. Lu, G. D. Hager, and E. Mjolsness. Fast and globally convergent pose estimation from video images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(6):610–622, 2000.

[5] B. Rosenhahn, C. Perwass, and G. Sommer. Pose estimation of 3d free-form contours. *Int. Journal of Computer Vision*, 62(3):267–289, 2005.

[6] B. Rosenhahn and G. Sommer. Pose estimation in conformal geometric algebra, part I: The stratification of mathematical spaces. *Journal of Mathematical Imaging and Vision*, 22:27–48, 2005.

[7] B. Rosenhahn and G. Sommer. Pose estimation in conformal geometric algebra, part II: Real-time pose estimation using extended feature concepts. *Journal of Mathematical Imaging and Vision*, 22:49–70, 2005.

[8] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *Int. J. Comput. Vision*, 13(2):119–152, 1994.