

Pose Estimation of Free-form Surface Models

Bodo Rosenhahn, Christian Perwass, Gerald Sommer

Institut für Informatik und Praktische Mathematik
Christian-Albrechts-Universität zu Kiel
Olshausenstr. 40, 24098 Kiel, Germany
{bro,chp,gs}@ks.informatik.uni-kiel.de

Abstract. In this article we discuss the 2D-3D pose estimation problem of 3D free-form surface models. In our scenario we observe free-form surface models in an image of a calibrated camera. Pose estimation means to estimate the relative position and orientation of the 3D object to the reference camera system. The object itself is modelled as a two-parametric surface model which is represented by Fourier descriptors. It enables a low-pass description of the surface model, which is advantageously applied to the pose problem. To achieve the combination of such a signal-based model within the geometry of the pose scenario, the conformal geometric algebra is used and applied.

1 Introduction

Pose estimation itself is one of the oldest computer vision problems. It is crucial for many computer and robot vision tasks. The problem is finding a rigid motion, which fits object models with image data. One main question is, how to represent objects, and the wide variety of literature deals with different entities concerning simple point or line correspondences up to general free-form contours. Pioneering work was done in the 80's and 90's by Lowe [7], Grimson [6] and others. These authors use point correspondences. More abstract entities can be found in [17, 2]. In the literature we find circles, cylinders, kinematic chains or other multi-part curved objects as entities. Works concerning free-form curves can be found in [4, 15]. Contour point sets, affine snakes, or active contours are used for visual servoing in these works. A free-form surface model can be represented for example as parametric form, implicit surface, superquadric, etc. An overview of free-form representations can e.g. be found in [3], though the focus of this work is on object recognition and not on pose estimation.

Pose estimation means to estimate the relative position and orientation of a 3D object to a reference camera system: We assume a 3D object model and the extracted silhouette of the object in an image of a calibrated camera. The aim is to find the rotation \mathbf{R} and translation \mathbf{t} of the object, which leads to the best fit of the reference model with the extracted silhouette. To relate 2D image information to 3D entities we interpret a point on the 2D silhouette as a projection ray in space, gained through projective reconstruction from the image point. This idea will be used to formulate the pose estimation problem in a 3D

scenario. Our recent work concentrates on modeling objects by using features of the object [12] (e.g. corners, edges, kinematic chains) and on modeling objects by using free-form contour models [11]. Instead, we now deal with 3D free-form surface models of objects. This is the next step of generalization of our existing algorithms and leads to the possibility of modeling more natural objects.

2 The pose problem in conformal geometric algebra

This section concerns the formalization of the free-form pose estimation problem in conformal geometric algebra. Geometric algebras are the language we use for the pose problem and the main argument for choosing this language is its possibility of coupling projective, kinematic and Euclidean geometry by using a conformal model. Besides, it enables a coordinate-free and dense symbolic representation. In this work we will only present basic principles of geometric algebras to give an idea of the rich properties of geometric algebras. A more detailed introduction to geometric algebras can be found in [13, 14].

The main idea of geometric algebras \mathcal{G} is to define a product on basis vectors which extends a linear vector space V of dimension n to a linear space of dimension 2^n with rich subspace structure. The elements are so-called multivectors as higher order algebraic entities in comparison to vectors of a vector space as first order entities. A geometric algebra is denoted as $\mathcal{G}_{p,q}$ with $n = p + q$. Here p and q indicate the numbers of basis vectors which square to $+1$ and -1 , respectively. The product defining a geometric algebra is called *geometric product* and is denoted by juxtaposition, e.g. uv for two multivectors u and v . Operations between multivectors can be expressed by special products, called *inner* \cdot , *outer* \wedge , *commutator* $\underline{\times}$ and *anticommutator* $\overline{\times}$ product. The idea behind conformal

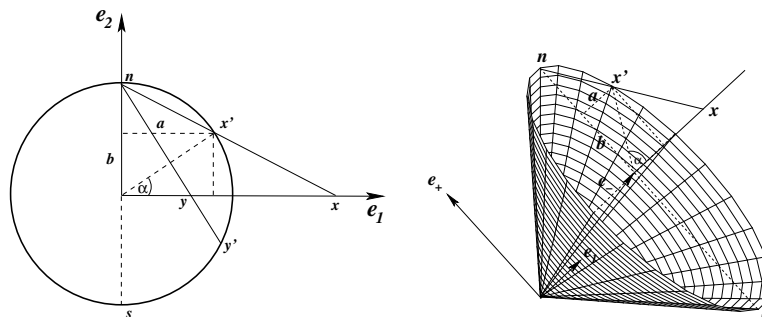


Fig. 1. Left: Visualization of a stereographic projection for the 1D case: Points on the line e_1 are projected on the (unit) circle and vice versa. Right: Visualization of the homogeneous model for a stereographic projection in the 1D case. All stereographic projected points are on a cone, which is a null cone in the Minkowski space.

geometry is to interpret points as *stereographically projected* points. This means

augmenting the dimension of space by one. The method used in a stereographic projection is visualized for the 1D case in the left image of figure 1: Points \mathbf{x} on the line \mathbf{e}_1 are mapped to points \mathbf{x}' on the unit circle by intersecting the line spanned by the *north pole* \mathbf{n} and \mathbf{x} with the circle. The basic formulas for projecting points in space on the hypersphere and vice versa are for example given in [10]. Using a homogeneous model for stereographic projected points means to augment the coordinate system by a further additional coordinate whose unit vector now squares to minus one. In 1D this leads to a cone in space, which is visualized in the right image of figure 1. This cone is spanned by the original coordinate system, an augmented dimension for the stereographic projection and an homogeneous dimension. This space is chosen to have a Minkowski metric and leads to a representation of any Euclidean point on a null cone (1D case) or a null hypercone (3D case). In [14] it is further shown that the conformal group of \mathbb{R}^n is isomorphic to the Lorentz group of $\mathbb{R}^{n+1,1}$ which has a spinor representation in $\mathcal{G}_{n+1,1}$. We will take advantage of both properties of the constructed embedding which are the representation of points as null-vectors and the spinor representation of the conformal group.

The conformal geometric algebra $\mathcal{G}_{4,1}$ (CGA) [8,13] is suited to describe conformal geometry. The point at infinity, $\mathbf{e} \simeq \mathbf{n}$, and the origin, $\mathbf{e}_0 \simeq \mathbf{s}$, are special elements of the representation which are used as basis vectors instead of \mathbf{e}_+ and \mathbf{e}_- because they define a null space in the conformal geometric algebra. A Euclidean point $\mathbf{x} \in \mathbb{R}^3$ can be represented as a point $\underline{\mathbf{x}}$ on the null cone by taking $\underline{\mathbf{x}} = \mathbf{x} + \frac{1}{2}\mathbf{x}^2\mathbf{e} + \mathbf{e}_0$. The multivector concepts of geometric algebras then allow to define entities like points, lines, planes, circles or spheres. Rotations are represented by rotors, $\mathbf{R} = \exp(-\frac{\theta}{2}\mathbf{l})$. The parameter of a rotor \mathbf{R} is the rotation angle θ applied to a unit bivector \mathbf{l} which represents the dual of the rotation axis. The rotation of an entity can be performed by its spinor product $\underline{\mathbf{X}}' = \mathbf{R}\underline{\mathbf{X}}\widetilde{\mathbf{R}}$. The multivector $\widetilde{\mathbf{R}}$ denotes the reverse of \mathbf{R} . A translation \mathbf{t} can be expressed in a similar manner with a translator, $\mathbf{T} = \exp(\frac{\mathbf{e}\mathbf{t}}{2})$. A rigid body motion can be expressed as a screw motion [9]. The motor \mathbf{M} describing a screw motion has the general form $\mathbf{M} = \exp(-\frac{\theta}{2}(\mathbf{n} + \mathbf{e}\mathbf{m}))$, with a unit bivector \mathbf{n} and an arbitrary 3D vector \mathbf{m} . The pair $(\theta\mathbf{n}, \theta\mathbf{m})$ in the exponential term is also called a *twist* [2].

Constraint equations for pose estimation

Now we start to express the 2D-3D pose estimation problem for pure point correspondences: *a transformed object point has to lie on a projection ray, reconstructed from an image point.* Let $\underline{\mathbf{X}}$ be a 3D object point given in CGA. The (unknown) transformation of the point can be described as $\mathbf{M}\underline{\mathbf{X}}\widetilde{\mathbf{M}}$. Let \mathbf{x} be an image point on a projective plane. The projective reconstruction of an image point in CGA can be written as $\underline{\mathbf{L}}_x = \mathbf{e} \wedge \mathbf{O} \wedge \mathbf{x}$. The line $\underline{\mathbf{L}}_x$ is calculated from the optical center \mathbf{O} , the image point \mathbf{x} and the vector \mathbf{e} as the point at infinity. The line $\underline{\mathbf{L}}_x$ is given in a Plücker representation. Collinearity can be described by the commutator product. Thus, the 2D-3D pose problem for a point $\underline{\mathbf{X}} \in \mathbb{R}^{4,1}$ can be formalized as constraint equation in CGA,

$$(\mathbf{M}\underline{\mathbf{X}}\widetilde{\mathbf{M}}) \times (\mathbf{e} \wedge \mathbf{O} \wedge \mathbf{x}) = 0.$$

Constraint equations which relate 2D image lines to 3D object points or 2D image lines to 3D object lines can be expressed in a similar manner. Note: The constraint equations in the unknown motor \mathbf{M} express a distance measure which has to be zero. The minimization of that distance leads to estimates of the pose.

Fourier descriptors in CGA

Fourier descriptors are often used for object recognition [5] and affine pose estimation [1] of closed contours. We are now concerned with the formalization of

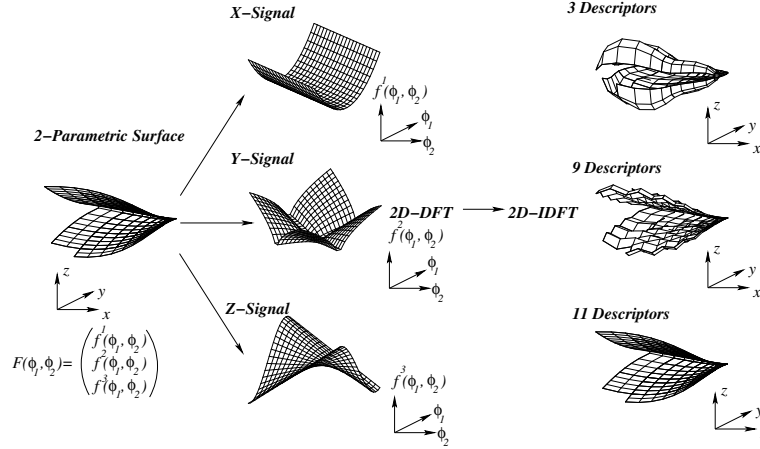


Fig. 2. Visualization of surface modeling and approximation by using three 2D Fourier descriptors.

3D Fourier descriptors in CGA. We assume a two-parametric surface of the form

$$F(\phi_1, \phi_2) = \sum_{i=1}^3 f^i(\phi_1, \phi_2) \mathbf{e}_i.$$

This means, we have three 2D functions $f^i(\phi_1, \phi_2) : \mathbb{R}^2 \rightarrow \mathbb{R}$ acting on the different base vectors \mathbf{e}_i . For a discrete number of sampled points, f_{n_1, n_2}^i , ($n_1 \in [-N_1, N_1]$; $n_2 \in [-N_2, N_2]$; $N_1, N_2 \in \mathbb{N}$) on the surface, we can now interpolate the surface by using a 2D discrete Fourier transform (2D-DFT) and then apply an inverse 2D discrete Fourier transform (2D-IDFT). The surface can therefore be approximated as a series expansion

$$\begin{aligned} F(\phi_1, \phi_2) &\simeq \sum_{i=1}^3 \sum_{k_1=-N_1}^{N_1} \sum_{k_2=-N_2}^{N_2} \mathbf{p}_{k_1, k_2}^i \exp\left(\frac{2\pi k_1 \phi_1}{2N_1 + 1} \mathbf{l}_i\right) \exp\left(\frac{2\pi k_2 \phi_2}{2N_2 + 1} \mathbf{l}_i\right) \\ &= \sum_{i=1}^3 \sum_{k_1=-N_1}^{N_1} \sum_{k_2=-N_2}^{N_2} \mathbf{R}_{1, i}^{k_1, \phi_1} \mathbf{R}_{2, i}^{k_2, \phi_2} \mathbf{p}_{k_1, k_2}^i \widetilde{\mathbf{R}}_{2, i}^{k_2, \phi_2} \widetilde{\mathbf{R}}_{1, i}^{k_1, \phi_1}. \end{aligned}$$

Here we have replaced the imaginary unit $i = \sqrt{-1}$ with three different rotation axes, represented by the bivectors \mathbf{l}_i , with $\mathbf{l}_i^2 = -1$. The complex Fourier series

coefficients are contained in the vectors \mathbf{p}_{k_1, k_2}^i that lie in the plane spanned by \mathbf{l}_i . We will call them phase vectors. These vectors can be obtained by a 2D-DFT of the sample points f_{n_1, n_2}^i on the surface,

$$\mathbf{p}_{k_1, k_2}^i = \frac{1}{(2N_1 + 1)(2N_2 + 1)} \sum_{n_1 = -N_1}^{N_1} \sum_{n_2 = -N_2}^{N_2} f_{n_1, n_2}^i \exp\left(-\frac{2\pi k_1 n_1}{2N_1 + 1} \mathbf{l}_i\right) \exp\left(-\frac{2\pi k_2 n_2}{2N_2 + 1} \mathbf{l}_i\right) \mathbf{e}_i.$$

This is visualized in figure 2: a two-parametric surface can be interpolated and approximated by using the estimated 2D Fourier descriptors.

Pose estimation of free-form surfaces

So far we have introduced the basic constraint equations for pose estimation and the surface representation of objects. We now continue with the algorithm for silhouette based pose estimation of surface models. In our scenario, we assume

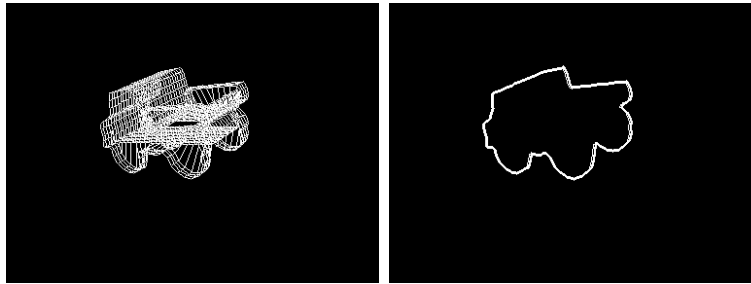


Fig. 3. Left: The projected surface model on a virtual image. Right: The estimated 3D silhouette of the surface model, back projected in an image.

to have extracted the silhouette of an object in an image. To compare points on the image silhouette with the surface model, the idea is to work with those points on the surface model which lie on the outline of a 2D projection of the object. This means we work with the 3D silhouette of the surface model with respect to the camera. To obtain this, the idea is to project the whole surface on a virtual image. Then the contour is calculated and from the image contour the 3D silhouette of the surface model is reconstructed. This is visualized in figure 3. The contour model is then used within our contour based pose estimation algorithm [11]. We are applying an ICP-algorithm [16]. Since the aspects of the surface model are changing during the ICP-cycles, a new silhouette will be estimated after each cycle to deal with occlusions within the surface model.

Solving a set of constraint equations for a free-form contour with respect to the unknown motor \mathbf{M} is a non-trivial task, since a motor corresponds to a polynomial of infinite degree. In [12] we presented a method which does not estimate the rigid body motion on the Lie group $SE(3)$, but the parameters which generate their Lie algebra $se(3)$, comparable to the ideas, presented in [2].

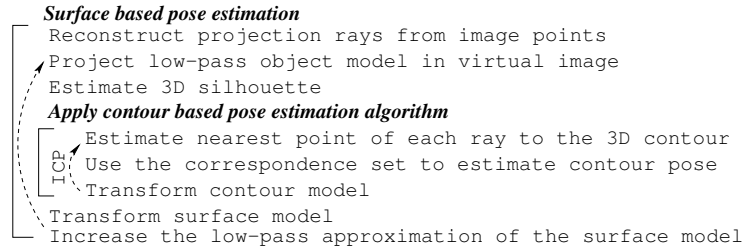


Fig. 4. The algorithm for pose estimation of surface models.

This means we linearize and iterate the equations. It corresponds to a gradient descent method in the 3D space. The algorithm for pose estimation of surface models is summarized in figure 4.

3 Experiments

Figure 5 shows different approximation levels of the surface model of a car. The approximations are achieved by using not all phase vectors of the surface model, but a subset leading to a low-pass description of the surface model. The object model itself consists of $69 \times 21 \approx 1450$ 3D points. In 3D it has a height, width and depth of $11cm \times 21cm \times 10cm$ and is used for the experiments in figures 6 and 7. The convergence behavior of the algorithm is shown in figure 6. As

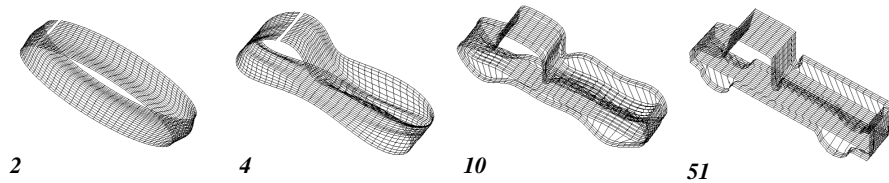


Fig. 5. Different approximation levels of the surface model. In the examples, 2, 4, 10 and 51 Fourier descriptors are used.

can be seen, we refine the pose results by using a low-pass approximation of the surface and by adding successively higher frequencies during the iteration. This is basically a multi-resolution method and helps to avoid getting stuck in local minima during the iteration.

Figure 7 shows different pose results obtained with our algorithm. Note, that our algorithms are even able to deal with non-homogeneous background and with camera noise. We implemented the sources in C++. The computing time of the

Module	Time (ms)	Module	Time (ms)	Module	Time (ms)
2D-DFT	700ms	Image processing	12 ms	ICP-cycle	50 ms
2D-IDFT	12ms - 700 ms	3D silhouette	20 ms		

Table 1. Time performance of the implemented modules. Note, the 2D-DFT and the 2D-IDFT are calculated once at the beginning of the image sequence.

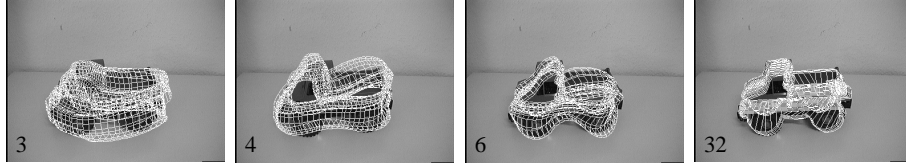


Fig. 6. Pose results of the low-pass contours during the ICP cycle.

different involved modules is summarized in table 1. These values are obtained with a Linux 2GHz machine. As can be seen, the 2D-DFT and 2D-IDFT are the bottleneck for the time performance. Therefore, the 2D-DFT and the 2D-IDFT is only estimated once at the beginning of the algorithm and the data is copied and transformed with the estimated rigid motion.

The overall computing times vary with the number of ICP-cycles and is for this object model around 400 ms for each image. We tested the algorithm on different image sequences containing up to 600 images.

4 Discussion

In this work we present a novel approach for free-form surface pose estimation. Free-form surfaces are modelled by three 2D Fourier descriptors and low-pass information is used for approximation. The estimated 3D silhouette is then combined with the pose estimation constraints. The coupling of geometry with signal theory is achieved by using the conformal geometric algebra. In this language we are able to fuse concepts, like complex numbers, Plücker lines, twists, Lie algebras and Lie groups in a compact manner. The experiments show the basic

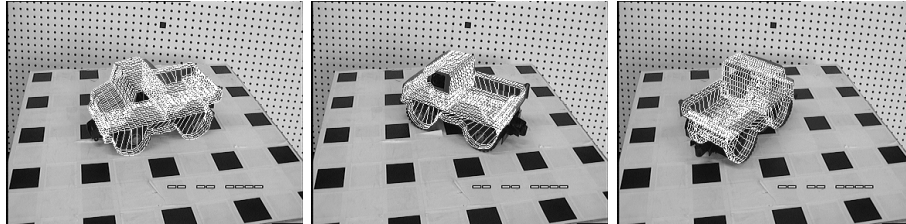


Fig. 7. Different pose results of the object model.

properties of the algorithm and future work will concentrate on collecting more experiences with this approach, making stability experiments, etc.

Acknowledgements

This work has been supported by DFG Graduiertenkolleg No. 357 and by EC Grant IST-2001-3422 (VISATEC).

References

1. Arbter K. and Burkhardt H. Ein Fourier-Verfahren zur Bestimmung von Merkmalen und Schätzung der Lageparameter ebener Raumkurven. *Informationstechnik*, Vol. 33, No. 1, pp. 19-26, 1991.
2. Bregler C. and Malik J. Tracking people with twists and exponential maps. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Santa Barbara, California, pp. 8-15 1998.
3. Campbell R.J. and Flynn P.J. A survey of free-form object representation and recognition techniques. *Computer Vision and Image Understanding (CVIU)*, No. 81, pp. 166-210, 2001.
4. Drummond T. and Cipolla R. Real-time tracking of multiple articulated structures in multiple views. In *6th European Conference on Computer Vision, ECCV 2000, Dublin, Ireland*, Part II, pp. 20-36, 2000.
5. Granlund G. Fourier preprocessing for hand print character recognition. *IEEE Transactions on Computers*, Vol. 21, pp. 195-201, 1972.
6. Grimson W. E. L. Object Recognition by Computer. *The MIT Press, Cambridge, MA*, 1990.
7. Lowe D.G. Solving for the parameters of object models from image descriptions. In *Proc. ARPA Image Understanding Workshop*, pp. 121-127, 1980.
8. Li H., Hestenes D. and Rockwood A. Generalized homogeneous coordinates for computational geometry. In [14], pp. 27-52, 2001.
9. Murray R.M., Li Z. and Sastry S.S. A Mathematical Introduction to Robotic Manipulation. *CRC Press*, 1994.
10. Needham T. Visual Complex Analysis. *Oxford University Press*, 1997
11. Rosenhahn B., Perwass Ch. and Sommer G. Pose estimation of 3D free-form contours in conformal geometry In *Proceedings of Image and Vision Computing (IVCNZ) D. Kenwright (Ed.)*, New Zealand, pp. 29-34, 2002.
12. Rosenhahn B. and Sommer G. Adaptive Pose Estimation for Different Corresponding Entities. In *Pattern Recognition, 24th DAGM Symposium, L. Van Gool (Ed.)*, Springer-Verlag, Berlin Heidelberg, LNCS 2449, pp. 265-273, 2002.
13. Rosenhahn B. and Sommer G. Pose Estimation in Conformal Geometric Algebra Part I: The stratification of mathematical spaces. Part II: Real-Time pose estimation using extended feature concepts. *Technical Report 0206*, University Kiel, 2002.
14. Sommer G., editor. Geometric Computing with Clifford Algebra. *Springer Verlag*, 2001.
15. Stark K. A method for tracking the pose of known 3D objects based on an active contour model. *Technical Report TUD / FI 96 10*, TU Dresden, 1996.
16. Zang Z. Iterative point matching for registration of free-form curves and surfaces. *IJCV: International Journal of Computer Vision*, Vol. 13, No. 2, pp. 119-152, 1999.
17. Zerroug, M. and Nevatia, R. Pose estimation of multi-part curved objects. *Image Understanding Workshop (IUW)*, pp. 831-835, 1996