GEOMETRIC ADVANCED TECHNIQUES FOR ROBOT GRASPING USING STEREOSCOPIC VISION

Julio Zamora-Esquivel and Eduardo Bayro-Corrochano
Department of Electrical Engineering and Computer Science
CINVESTAV, unidad Guadalajara. Jalisco, Mexico

Keywords: Conformal Geometry, Kinematics, Grasping, Tracking.

Abstract: In this paper the authors propose geometric techniques to deal with the problem of grasping objects relying on their mathematical models. For that we use the geometric algebra framework to formulate the kinematics of a three finger robotic hand. Our main objective is by formulating a kinematic control law to close the loop between perception and actions. This allows us to perform a smooth visually guided object grasping action.

1 INTRODUCTION

In this work the authors show how to obtain a feasible grasping strategy based on the mathematical model of the object and the manipulator. In order to close the loop between perception and action we estimate the pose of the object and the robot hand. A control law is also proposed using the mechanical Jacobian matrix computed using the lines of the axis of the Barrett hand. Conformal geometric algebra has been used within this work instead of the projective approach (Ruf, 2000) due to the advantages which are provided by this mathematical framework in the process of modeling of the mechanical structures like the one of the Barrett Hand.

In our approach first we formulate the inverse kinematics of the robot hand and analyze the object models in order to identify the grasping constraints. This takes into account suitable contact points between object and robot hand. Finally a control law to close the perception and action loop is proposed. In the experimental analyzes we present a variety of real grasping situations.

2 GEOMETRIC ALGEBRA

Let $\mathcal{G}_n$ denote the geometric algebra of n-dimensions, this is a graded linear space. As well as vector addition and scalar multiplication we have a non-commutative product which is associative and distributive over addition – this is the *geometric* or *Clifford product*.

The inner product of two vectors is the standard *scalar* product and produces a scalar. The outer or wedge product of two vectors is a new quantity which we call a *bivector*. Thus, $b \wedge a$ will have the opposite orientation making the wedge product anti-commutative. The outer product is immediately generalizable to higher dimensions – for example, $(a \wedge b) \wedge c$, a *trivector*, is interpreted as the oriented volume formed by sweeping the area $a \wedge b$ along vector c. The outer product of k vectors is a k-vector or k-blade, and such a quantity is said to have *grade k*. A *multivector* (linear combination of objects of different type) is *homogeneous* if it contains terms of only a single grade.

We will specify a geometric algebra $\mathcal{G}_n$ of the n dimensional space by $\mathcal{G}_{p,q,r}$, where p, q and r stand for the number of basis vector which squares to 1, -1 and 0 respectively and fulfill $n = p + q + r$.

We will use e_i to denote the vector basis i. In a Geometric algebra $G_{p,q,r}$, the geometric product of two basis vector is defined as

$$e_i e_j = \begin{cases} 1 & for \quad i = j \in 1, \cdots, p \\ -1 & for \quad i = j \in p+1, \cdots, p+q \\ 0 & for \quad i = j \in p+q+1, \cdots, p+q+r. \\ e_i \wedge e_j & for \quad i \neq j \end{cases}$$

This leads to a basis for the entire algebra:

$$\{1\}, \{e_i\}, \{e_i \wedge e_j\}, \{e_i \wedge e_j \wedge e_k\}, \ldots, \{e_1 \wedge e_2 \wedge \ldots \wedge e_n\} \qquad (1)$$

Any multivector can be expressed in terms of this basis.

3 CONFORMAL GEOMETRY

Geometric algebra $G_{4,1}$ can be used to treat conformal geometry in a very elegant way. To see how this is possible, we follow the same formulation presented

Zamora-Esquivel J. and Bayro-Corrochano E. (2007).
GEOMETRIC ADVANCED TECHNIQUES FOR ROBOT GRASPING USING STEREOSCOPIC VISION.
In *Proceedings of the Fourth International Conference on Informatics in Control, Automation and Robotics*, pages 175-182
DOI: 10.5220/0001627701750182

in (H. Li, 2001) and show how the Euclidean vector space $\mathbb{R}^3$ is represented in $\mathbb{R}^{4,1}$. This space has an orthonormal vector basis given by $\{e_i\}$ and $e_{ij} = e_i \wedge e_j$ are bivectorial basis and e_{23}, e_{31} and e_{12} correspond to the Hamilton basis. The unit Euclidean pseudo-scalar $I_e := e_1 \wedge e_2 \wedge e_3$, a pseudo-scalar $I_c := I_e E$ and the bivector $E := e_4 \wedge e_5 = e_4 e_5$ are used for computing the inverse and duals of multivectors.

3.1 The Stereographic Projection

The conformal geometry is related to a stereographic projection in Euclidean space. A stereographic projection is a mapping taking points lying on a hypersphere to points lying on a hyperplane. In this case, the projection plane passes through the equator and the sphere is centered at the origin. To make a projection, a line is drawn from the north pole to each point on the sphere and the intersection of this line with the projection plane constitutes the stereographic projection.

For simplicity, we will illustrate the equivalence between stereographic projections and conformal geometric algebra in $\mathbb{R}^1$. We will be working in $\mathbb{R}^{2,1}$ with the basis vectors $\{e_1, e_4, e_5\}$ having the usual properties. The projection plane will be the x-axis and the sphere will be a circle centered at the origin with unitary radius.

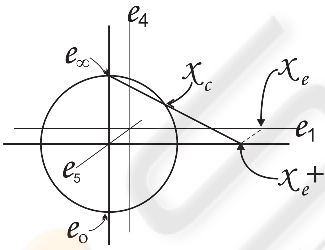

Figure 1: Stereographic projection for 1-D.

Given a scalar x_e representing a point on the x-axis, we wish to find the point x_c lying on the circle that projects to it (see Figure 1). The equation of the line passing through the north pole and x_e is given by $f(x) = -\frac{1}{x_e}x + 1$ and the equation of the circle $x^2 + f(x)^2 = 1$. Substituting the equation of the line on the circle, we get the point of intersection x_c, which can be represented in homogeneous coordinates as the vector

$$x_c = 2\frac{x_e}{x_e^2+1}e_1 + \frac{x_e^2-1}{x_e^2+1}e_4 + e_5. \qquad (2)$$

From (2) we can infer the coordinates on the circle for the point at infinity as

$$e_\infty = \lim_{x_e \to \infty} \{x_c\} = e_4 + e_5, \qquad (3)$$

$$e_o = \frac{1}{2}\lim_{x_e \to 0} \{x_c\} = \frac{1}{2}(-e_4 + e_5), \qquad (4)$$

Note that (2) can be rewritten to

$$x_c = x_e + \frac{1}{2}x_e^2 e_\infty + e_o, \qquad (5)$$

3.2 Spheres and Planes

The equation of a sphere of radius ρ centered at point $p_e \in \mathbb{R}^n$ can be written as $(x_e - p_e)^2 = \rho^2$. Since $x_c \cdot y_c = -\frac{1}{2}(\mathbf{x}_e - \mathbf{y}_e)^2$ and $x_c \cdot e_\infty = -1$ we can factor the expression above to

$$x_c \cdot (p_c - \frac{1}{2}\rho^2 e_\infty) = 0. \qquad (6)$$

Which finally yields the simplified equation for the sphere as $s = p_c - \frac{1}{2}\rho^2 e_\infty$. Alternatively, the dual of the sphere is represented as 4-vector $s^* = sI_c$. The sphere can be directly computed from four points as

$$s^* = x_{c_1} \wedge x_{c_2} \wedge x_{c_3} \wedge x_{c_4}. \qquad (7)$$

If we replace one of these points for the point at infinity we get the equation of a plane

$$\pi^* = x_{c_1} \wedge x_{c_2} \wedge x_{c_3} \wedge e_\infty. \qquad (8)$$

So that π becomes in the standard form

$$\pi = I_c \pi^* = n + d e_\infty \qquad (9)$$

Where n is the normal vector and d represents the Hesse distance.

3.3 Circles and Lines

A circle z can be regarded as the intersection of two spheres s_1 and s_2 as $z = (s_1 \wedge s_2)$. The dual form of the circle can be expressed by three points lying on it

$$z^* = x_{c_1} \wedge x_{c_2} \wedge x_{c_3}. \qquad (10)$$

Similar to the case of planes, lines can be defined by circles passing through the point at infinity as:

$$L^* = x_{c_1} \wedge x_{c_2} \wedge e_\infty. \qquad (11)$$

The standard form of the line can be expressed by

$$L = l + e_\infty (t \cdot l), \qquad (12)$$

the line in the standard form is a bivector, and it has six parameters (Plucker coordinates), but just four degrees of freedom.

4 DIRECT KINEMATICS

The direct kinematics involves the computation of the position and orientation of the end-effector given the parameters of the joints. The direct kinematics can be easily computed given the lines of the axes of screws.

4.1 Rigid Transformations

We can express rigid transformations in conformal geometry carrying out reflections between planes.

4.1.1 Reflection

The reflection of conformal geometric entities help us to do any other transformation. The reflection of a point x with respect to the plane π is equal x minus twice the direct distance between the point and plane as shown in figure 2.

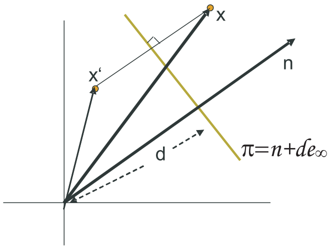

Figure 2: Reflection of a point x respect to the plane π.

For any geometric entity Q, the reflection respect to the plane π is given by

$$Q' = \pi Q \pi^{-1} \tag{13}$$

4.1.2 Translation

The translation of conformal entities can be done by carrying out two reflections in parallel planes π_1 and π_2 see the figure (3), that is

$$Q' = \underbrace{(\pi_2\pi_1)}_{T_a} Q \underbrace{(\pi_1^{-1}\pi_2^{-1})}_{\widetilde{T_a}} \tag{14}$$

$$T_a = (n+de_\infty)n = 1+\frac{1}{2}ae_\infty = e^{-\frac{a}{2}e_\infty} \tag{15}$$

With $a = 2dn$.

4.1.3 Rotation

The rotation is the product of two reflections between nonparallel planes, (see figure (4))

$$Q' = \underbrace{(\pi_2\pi_1)}_{R_\theta} Q \underbrace{(\pi_1^{-1}\pi_2^{-1})}_{\widetilde{R_\theta}} \tag{16}$$

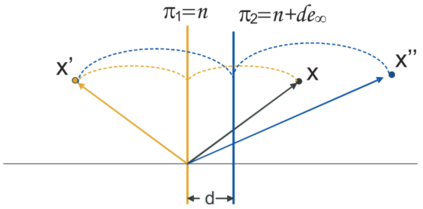

Figure 3: Reflection about parallel planes.

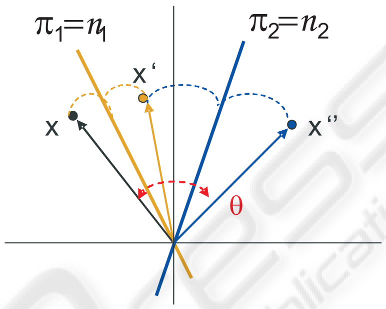

Figure 4: Reflection about nonparallel planes.

Or computing the conformal product of the normals of the planes.

$$R_\theta = n_2n_1 = Cos(\frac{\theta}{2}) - Sin(\frac{\theta}{2})l = e^{-\frac{\theta}{2}l} \tag{17}$$

With $l = n_2 \wedge n_1$, and θ twice the angle between the planes π_2 and π_1. The screw motion *called motor* in (Bayro-Corrochano and Kahler, 2000) related to an arbitrary axis L is $M = TR\widetilde{T}$

$$Q' = \underbrace{(TR\widetilde{T})}_{M_\theta} Q \underbrace{((T\widetilde{R}\widetilde{T}))}_{\widetilde{M_\theta}} \tag{18}$$

$$M_\theta = TR\widetilde{T} = Cos(\frac{\theta}{2}) - Sin(\frac{\theta}{2})L = e^{-\frac{\theta}{2}L} \tag{19}$$

4.2 Kinematic Chains

The direct kinematics for serial robot arms is a succession of motors and it is valid for points, lines, planes, circles and spheres.

$$Q' = \prod_{i=1}^{n} M_i Q \prod_{i=1}^{n} \widetilde{M}_{n-i+1} \tag{20}$$

5 BARRETT HAND DIRECT KINEMATICS

The direct kinematics involves the computation of the position and orientation of the end-effector given the parameters of the joints. The direct kinematics can be easily computed given the lines of the axes of screws.

In order to explain the kinematics of the Barrett hand, we show the kinematics of one finger. In this example we will assume that the finger is totally extended. Note that such a hypothetical position is not reachable in normal operation, but this simplifies the explanation.

We initiated denoting some points on the finger which help to describe their position.

$$\begin{aligned} x_{1o} &= A_w e_1 + A_1 e_2 + D_w e_3, & (21)\\ x_{2o} &= A_w e_1 + (A_1 + A2) e_2 + D_w e_3, & (22)\\ x_{3o} &= A_w e_1 + (A_1 + A_2 + A_3) e_2 + D_w e_3. & (23) \end{aligned}$$

The points x_{1o}, x_{2o} and x_{3o} describe the position of each union and the end of the finger in the Euclidean space, see the figure 5.

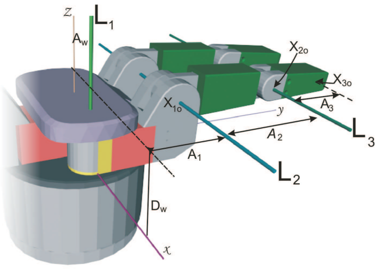

Figure 5: Barrett hand hypotetical position.

Having defined these points it is quite simple to calculate the axes,which will be used as motor's axis.

$$\begin{aligned} L_{1o} &= -A_w(e_2 \wedge e_\infty) + e_{12}, & (24)\\ L_{2o} &= (x_{1o} \wedge e_1 \wedge e_\infty) I_c, & (25)\\ L_{3o} &= (x_{2o} \wedge e_1 \wedge e_\infty) I_c, & (26) \end{aligned}$$

when the hand is initialized the fingers moves away to home position, this is $\Phi_2 = 2.46^o$ in union two and $\Phi_3 = 50^o$ degrees in union three. In order to move the finger from this hypothetical position to its home position the appropriate transformation need to be obtained.

$$\begin{aligned} M_{2o} &= cos\,(\Phi_2/2) - sin(\Phi_2/2) L_{2o}, & (27)\\ M_{3o} &= cos\,(\Phi_3/2) - sin(\Phi_3/2) L_{3o}, & (28) \end{aligned}$$

Having obtained the transformations, then we apply them to the points and lines to them that must move.

$$\begin{aligned} x_2 &= M_{2o} x_{2o} \widetilde{M}_{2o}, & (29)\\ x_3 &= M_{2o} M_{3o} x_{3o} \widetilde{M}_{3o} \widetilde{M}_{2o}, & (30)\\ L_3 &= M_{2o} L_{3o} \widetilde{M}_{2o}. & (31) \end{aligned}$$

The point $x_1 = x_{1o}$ is not affected by the transformation, as are for the lines $L_1 = L_{1o}$ and $L_2 = L_{2o}$ see figure 6.

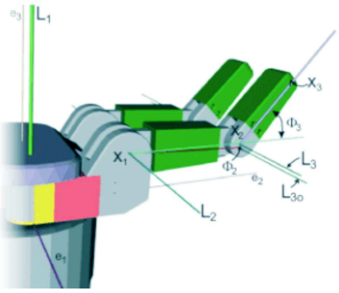

Figure 6: Barrett hand at home position.

Since the rotation angle of both axis L_2 and L_3 are related, we will use fractions of the angle q_1 to describe their individual rotation angles. The motors of each joint are computed using $\frac{2}{35}2q_4$ to rotate around L_1, $\frac{1}{125}q_1$ around L_2 and $\frac{1}{375}q_1$ around L_3, the angles coefficients were taken from the Barrett hand user manual.

$$\begin{aligned} M_1 &= cos(q_4/35) + sin(q_4/35) L_1, & (32)\\ M_2 &= cos(q_1/250) - sin(q_1/250) L_2, & (33)\\ M_3 &= cos(q_1/750) - sin(q_1/750) L_3. & (34) \end{aligned}$$

The position of each point is related to the angles q_1 and q_4 as follows:

$$\begin{aligned} x'_1 &= M_1 x_1 \widetilde{M}_1, & (35)\\ x'_2 &= M_1 M_2 x_2 \widetilde{M}_2 \widetilde{M}_1, & (36)\\ x'_3 &= M_1 M_2 M_3 x_3 \widetilde{M}_3 \widetilde{M}_2 \widetilde{M}_1, & (37)\\ L'_3 &= M_1 M_2 L_3 \widetilde{M}_2 \widetilde{M}_1, & (38)\\ L'_2 &= M_1 L_2 \widetilde{M}_1. & (39) \end{aligned}$$

Since we already know x'_3, L'_1, L'_2 and L'_3 we can calculate the speed of the end of the finger using

$$\dot{X}'_3 = X'_3 \cdot \left(-\frac{2}{35} L'_1 \dot{q}_4 + \frac{1}{125} L'_2 \dot{q}_1 + \frac{1}{375} L'_3 \dot{q}_1\right). \quad (40)$$

6 POSE ESTIMATION

There are many approaches to solve the pose estimation problem ((Hartley and Zisserman, 2000)). In our approach we project the known mathematical model of the object on the camera's image. This is possible because after calibration we know the intrinsic parameters of the camera, see fig 7. The image of the mathematical projected model is compared with the image of the segmented object. If we find a match between them, then this means that the mathematical object is placed in the same position and orientation as the real object. Otherwise we follow a descendant gradient

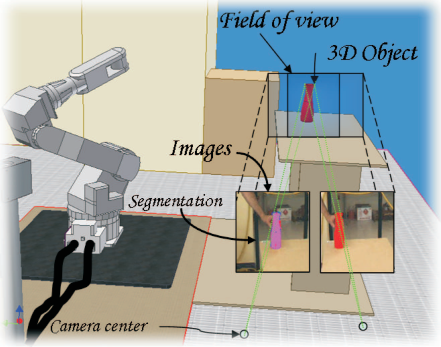

Figure 7: Mathematical model of the object.

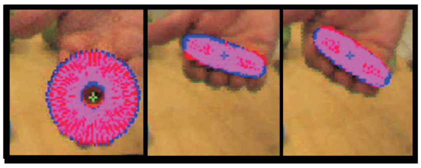

Figure 8: Pose estimation of a disk with a fixed camera.

based algorithm to rotate and translate the mathematical model in order to reduce the error between them. This algorithm runs very fast

Figure 8 shows the pose estimation result. In this case we have a maximum error of 0.4^o in the orientation estimation and 5mm of maximum error in the position estimation of the object. The problem becomes more difficult to solve when the stereoscopic system is moving. Figure 9 shows how well the stereo system track the object. If we want to know the real object's position with respect to the world coordinate system, of course we must know the extrinsic camera's parameters. Figure 10 illustrates the object's position and orientation with respect to the robot's hand. In the upper row of this figure we can see an augmented reality position sequence of the object. This shows that we can add the mathematical object in the real image. Furthermore, in the second row of the same image we can see the virtual reality pose estimation result.

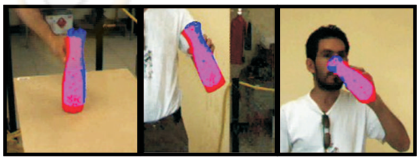

Figure 9: Pose estimation of a recipient.

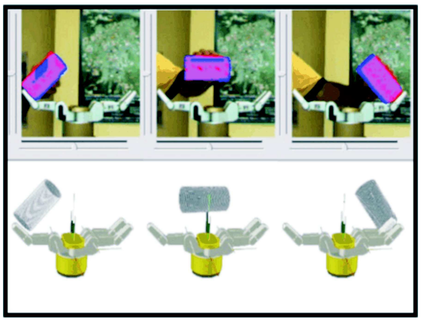

Figure 10: Object presented in augmented and virtual reality.

7 GRASPING THE OBJECTS

Considering that in using cameras we can only see the surface of the observed objects, in this work we consider them as bidimensional surfaces which are embed in a 3D space, and are described by the following function

$$\vec{H}(s,t) = h_x(s,t)e_1 + h_y(s,t)e_2 + h_z(s,t)e_3, \qquad (41)$$

where s and t are real parameters in the range $[0,1]$. Such parametrization allows us to work with different objects like points, conics, quadrics, or even more complex real objects like cups, glasses, etc.

Table 1: Functions of some objects.

Particle	$\vec{H} = 3e_1 + 4e_2 + 5e_3$
Cylinder	$\vec{H} = cos(t)e_1 + sin(t)e_2 + se_3$
Plane	$\vec{H} = te_1 + se_2 + (3s + 4t + 2)e_3$

There are many styles of grasping, however we are taking into account only three principal styles. Note that also for each style of grasping there are many possible solutions, for another approach see (Ch Borst and Hirzinger, 1999).

7.1 Style of Grasp One

Since our objective is to grasp such objects with the Barrett Hand, we must consider that it has only three fingers, so the problem consists in finding three points of grasping for which the system is in equilibrium by holding; this means that the sum of the forces are equal to zero, and also the sum of the moments.

We know the surface of the object, so we can compute its normal vector in each point using

$$N(s,t) = \left(\frac{\partial \vec{H}(s,t)}{\partial s} \wedge \frac{\partial \vec{H}(s,t)}{\partial t} \right) I_e. \qquad (42)$$

In surfaces with low friction the value of F tends to its projection over the normal ($F \approx F_n$). To maintain equilibrium, the sum of the forces must be zero $\sum_{i=1}^{3} \|F_n\| N(s_i,t_i) = 0$, (Fig. 11).

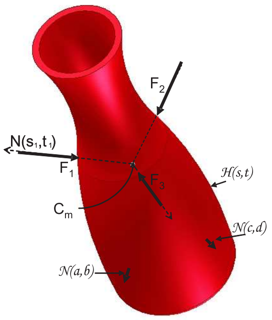

Figure 11: Object and his normal vectors.

This fact restricts the points over the surface in which the forces can be applied. This number of points is more reduced if we consider that the forces over the object are equal.

$$\sum_{i=1}^{3} N(s_i,t_i) = 0. \tag{43}$$

Additionally, in order to maintain the equilibrium of the system, it must be accomplished that the sum of the moments is zero

$$\sum_{i=1}^{3} H(s,t) \wedge N(s,t) = 0. \tag{44}$$

The points on the surface with the maximum and minim distance to the mass's center of the object fulfill $H(s,t) \wedge N(s,t) = 0$. The normal vector in such points crosses the center of mass (C_m) and it does not produce any moment. Before determining the external and internal points, we must compute the center of mass as

$$C_m = \int_0^1 \int_0^1 \vec{H}(s,t) ds dt \tag{45}$$

Once C_m is calculated we can establish the next restriction

$$(H(s,t) - C_m) \wedge N(s,t) = 0. \tag{46}$$

The values s and t satisfying (46), form a subspace and they fulfill that $H(s,t)$ are critical points on the surface (being maximums, minimums or inflections)

The constraint imposing that the three forces must be equal is hard to fulfill because it implies that the three points must be symmetric with respect to the mass center. When such points are not present, we can relax the constraint to allow that only two forces are equal in order to fulfill the hand's kinematics equations. Then, the normals $N(s_1,t_1)$ and $N(s_2,t_2)$ must be symmetric with respect to $N(s_3,t_3)$

$$N(s_3,t_3)N(s_1,t_1)N(s_3,t_3)^{-1} = N(s_2,t_2) \tag{47}$$

7.2 Style of Grasp Two

In the previous style of grasping three points of contact were considered. In this section we are taking into account a greater number of contact points, this fact generates a style of grasping that take the objects more secure. To increment the number of contact points is taken into account the base of the hand.

Since the object is described by the equation $H(s,t)$ it is possible to compute a plane π_b that divides the object in the middle, this is possible using lineal regression and also for the principal axis L_p. See figure 12.

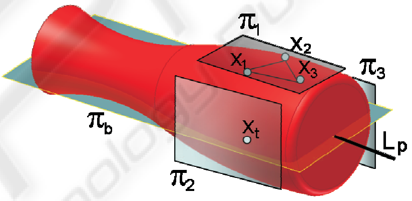

Figure 12: Planes of the object.

One Select only the points from locations with normal parallels to the plane π_b

$$N(s,t) \wedge \pi_b \approx 0 \tag{48}$$

Now we chose three points separated by 25 mm to generate a plane in the object. In this style of grasping the position of the hand relative to the object is trivial, because we just need to align the center of these points with the center of the hand's base. Also the orientation is the normal of the plane $\pi_1 = x_1 \wedge x_2 \wedge x_3 \wedge e_\infty$.

7.3 Style of Grasp Three

In this style of grasping the forces F_1, F_2 and F_3 do not intersect the mass center. They are canceled by symmetry, because the forces are parallel.

$$N(s_3,t_3)F_3 = N(s_1,t_1)F_1 + N(s_2,t_2)F_2. \tag{49}$$

Also the forces F_1, F_2 and F_3 are in the plane π_b and they are orthogonal to the principal axis L_p ($\pi_b = L_p \cdot N(s,t)$) as you can see in the figure 13.

A new restriction is then added to reduce the subspace of solutions

$$F_3 = 2F_1 = 2F_2, \tag{50}$$

$$N(s_1,t_1) = N(s_2,t_2) = -N(s_3,t_3). \tag{51}$$

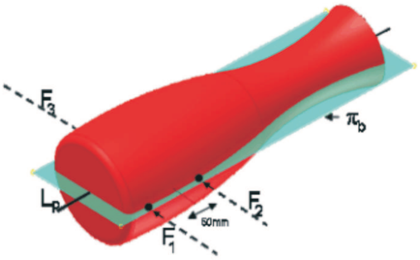

Figure 13: Forces of grasping.

Finally the direct distance between the parallels apply to x_1 and x_2 must be equal to 50 mm and between x_1, x_2 to x_3 must be equal to 25 mm.

Now we search exhaustively three points changing s_i and t_i. Figure 14 shows the simulation and result of this grasping algorithm. The position of the object relative to the hand must be computed using a coordinate frame in the object and other in the hand.

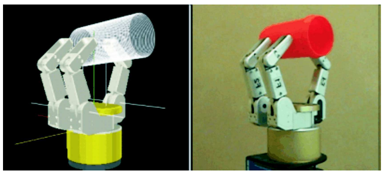

Figure 14: Simulation and result of the grasping.

8 TARGET POSE

Once the three grasping points ($P_1 = H(s_1,t_1)$, $P_2 = H(s_2,t_2)$, $P_3 = H(s_3,t_3)$) are calculated, for each finger it is really easy to determine the angles at the joints. To determine the angle of the spread ($q_4 = \beta$), we use

$$cos\beta \quad = \quad \frac{(p_1 - C_m)\cdot(C_m - p_3)}{|p_1 - c_m|\,|C_m - p_3|}. \tag{52}$$

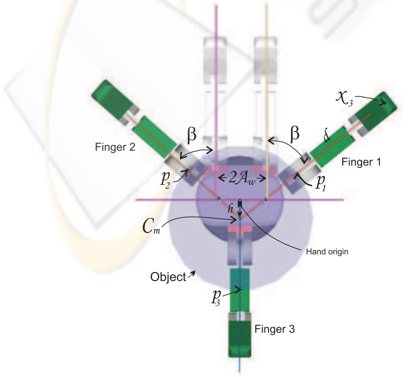

Figure 15: Object's position relative to the hand.

To calculate each one of the finger angles, we determine its elongation as

$$x_3' \cdot e_2 \quad = \quad |(p_1 - C_m)| - \frac{A_w}{\sin(\beta)} - A_1, \tag{53}$$

$$x_3' \cdot e_2 \quad = \quad |(p_2 - C_m)| - \frac{A_w}{\sin(\beta)} - A_1, \tag{54}$$

$$x_3' \cdot e_2 \quad = \quad |(p_3 - C_m)| + h - A_1, \tag{55}$$

where $x_3' \cdot e_2$ determines the opening distance of the finger

$$x_3' \cdot e_2 = \quad (M_2 M_3 x_3 \widetilde{M}_3 \widetilde{M}_2) \cdot e_2 \tag{56}$$

$$\begin{aligned} x_3' \cdot e_2 = \quad & A_1 + A_2 \cos\left(\tfrac{1}{125} q + I_2\right) + \\ & + A_3 \cos\left(\tfrac{4}{375} q + I_2 + I_3\right). \end{aligned} \tag{57}$$

Solving for the angle q we have the opening angle for each finger. These angles are computed off line for each style of grasping of each object. They are the target in the velocity control of the hand.

8.1 Object Pose

We must find the transformation M which allows us to put the hand in a such way that each finger-end coincides with the corresponding contact point. For the sake of simplicity transformation M is divided in three transformations (M_1, M_2, M_3). With the same purpose we label the finger ends as X_1, X_2 and X_3, and the contact points as P_1, P_2 and P_3.

The first transformation M_1 is the translation between the object and the hand, which is equal to the directed distance between the centers of the circles called $Z_h^* = X_1 \wedge X_2 \wedge X_3$ y $Z_o^* = P_1 \wedge P_2 \wedge P_3$, and it can be calculated as

$$M_1 = e^{-\frac{1}{2}\left(\frac{Z_h^*}{Z_h^* \wedge e_\infty} \wedge \frac{Z_o^*}{Z_o^* \wedge e_\infty} \wedge e_\infty\right) I_c}. \tag{58}$$

The second transformation allows the alignment of the planes $\pi_h^* = Z_h^* = X_1 \wedge X_2 \wedge X_3 \wedge e_\infty$ and $\pi_o^* = Z_o^* \wedge e_\infty$, which are generated by the new points of the hand and the object. This transformation is calculated as $M_2 = e^{-\frac{1}{2}\pi_h \wedge \pi_o}$. The third transformation allows that the points overlap and this can be calculated using the planes $\pi_1^* = Z_o \wedge X_3 \wedge e_\infty$ and $\pi_2^* = Z_o \wedge P_3 \wedge e_\infty$, which are generated by the circle's axis and any of the points $M_3 = e^{-\frac{1}{2}\pi_1 \wedge \pi_2}$.

These transformations define also the pose of the object relative to the hand. They are computed off line in order to know the target position and orientation of the object with respect to the hand, it will be used to design a control law for visually guided grasping

9 VISUALLY GUIDED GRASPING

Once the target position and orientation of the object is known for each style of grasping and the hand's posture (angles of joints), it is possible to write a control law using this information and the equation of differential kinematics of the hand that it allows by using visual guidance to take an object.
Basically the control algorithm takes the pose of the object estimated as shown in the Section 6 and compares with the each one of the target poses computed in the Section 8 in order to choose as the target the closest pose, in this way the style of grasping is automatically chosen.

Once the style of grasping is chosen and target pose is known, the error ε between the estimated and computed is used to compute the desired angles in the joints of the hand

$$\alpha_d = \alpha_t e^{-\varepsilon^2} + (1 - e^{-\varepsilon^2})\alpha_a \tag{59}$$

where α_d is the desired angle of the finger, α_t is the target angle computed in the section 8 and α_a is the actual angle of the finger. Now the error between the desired and the actual position is used to compute the new joint angle using the equation of differential kinematics of the Barrett hand given in the Section 5.

9.1 Results

Next we show the results of the combination of the algorithms of pose estimation, visual control and grasping to create a new algorithm for visually guided grasping. In the Figure 16 a sequence of images of the grasping is presented. When the bottle is approached by the hand the fingers are looking for a possible point of grasp.

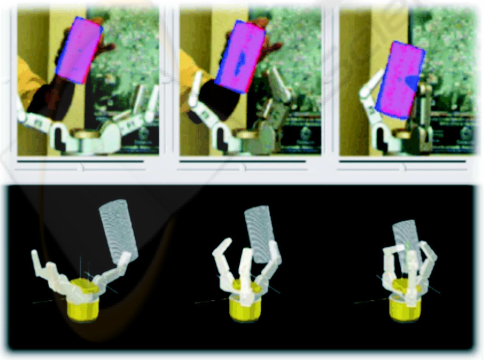

Figure 16: Visually guided grasping.

Now we can change the object or the pose of the object and the algorithm is computing a new behavior of grasping. The figure (17) shows a sequence of images changing the pose of the object.

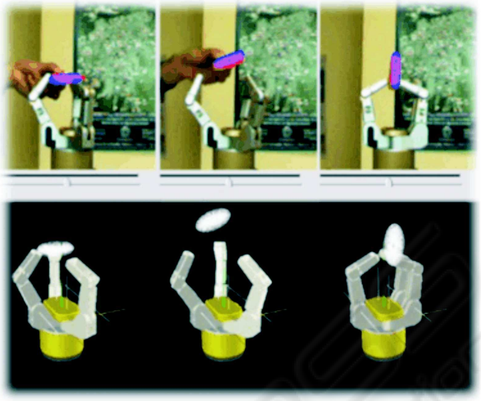

Figure 17: Changing the object's pose.

10 CONCLUSION

In this paper the authors used conformal geometric algebra to formulate grasping techniques. Using stereo vision we are able to detect the 3D pose and the intrinsic characteristics of the object shape. Based on this intrinsic information we developed feasible grasping strategies.

This paper emphasizes the importance of the development of algorithms for perception and grasping using a flexible and versatile mathematical framework

REFERENCES

Bayro-Corrochano, E. and Kahler, D. (2000). Motor algebra approach for computing the kinematics of robot manipulators. In *Journal of Robotic Systems*. 17(9):495-516.

Ch Borst, M. F. and Hirzinger, G. (1999). A fast and robust grasp planner for arbitrary 3d objects. In *ICRA99, International Conference on Robotics and Automation*. pages 1890-1896.

H. Li, D. Hestenes, A. R. (2001). *Generalized Homogeneous coordinates for computational geometry*. pages 27-52, in ((Somer, 2001)).

Hartley and Zisserman, A. (2000). *Multiple View Geometry in Computer Vision*. Cambridge University Press, UK, 1st edition.

Ruf, A. (2000). Closing the loop between articulated motion and stereo vision: a projective approach. In *PhD. Thesis, INP, GRENOBLE*.

Somer, G. (2001). *Geometric Computing with Clifford Algebras*. Springer-Verlag, Heidelberg, 2nd edition.