Proceedings of the 2003 IEEE/RSJ
Intl. Conference on Intelligent Robots and Systems
Las Vegas, Nevada · October 2003

An auto-epipolar strategy for mobile robot visual servoing

Jacopo Piazzi Domenico Prattichizzo
Dipartimento di Ingegneria dell'Informazione - Università di Siena
Via Roma, 56 - 53100 Siena, Italy
email: {piazzi,prattichizzo}@dii.unisi.it

I. Abstract

A novel visual servoing method is presented. The algorithm works for both apparent contours and point features and does not require any information about the internal camera parameters. The proposed visual servoing is based on the epipolar geometry and exploits the auto-epipolar property, a special configuration for the epipoles which occurs when the desired and current views undergo a pure translation. This occurrence is simply detectable from special line conditions on the overlapped current and desired images. Experiments are presented to validate the proposed visual servoing.

II. Introduction

Visual Servoing consists in compensating the robot position error exploiting a visual feedback, the control goals and the feedback law are directly designed in the image domain. In [13] the authors present a classification of visual servoing systems. The approach used in this paper is image-based (IBVS) and as image features the epipolar lines, obtained from corresponding points, are considered.

In visual servoing the angular and translational velocities of the camera are related to the motion of the point features through the Image Jacobian [11], [8], [10], [2], [14], [13]. An interesting approach, referred to as 2D 1/2 visual servoing, using point matches has been presented in [16] where the authors combine a partial pose estimation with imagebased servoing. Correspondences between points in the current and target images are also exploited for recovering the epipolar geometry [17]. In [1] feature points are used to estimate the fundamental matrix and the relative orientation and translation between the cameras. Taylor *et al.* [24] present an approach witch uses the epipolar geometry to do a partial pose reconstruction. In [21] the epipolar geometry is used to steer the feature points to a well defined configuration. Otherwise, as pointed out in [19], if the scene does not have any noticeable texture and only smooth surfaces are present, the object profiles are the only information available to estimate the structure of the surface and the motion of the camera. Cipolla *et al.* [3], [18] use apparent contours and profiles to recover camera motion. In [23] it was shown how the viewer motion can be computed from the constraints on the camera motion and the epipolar tangency points. In [4] the authors propose an algorithm for mobile robot planar navigation exploiting the epipolar geometry and some special symmetry conditions of epipoles.

In this paper, a new visual servoing based on epipolar geometry is proposed. It consists in exploiting the epipolar geometry between two views to steer the camera-robot to the desired configuration. This paper builds upon previous contributions [4], [5] and presents the auto-epipolar property [7], which is a collinearity property, as an important breakthrough of the visual servoing design based on epipoles. The proposed visual servoing does not need any knowledge on the camera calibration parameters and on the 3D scene structure. Moreover, it can work indifferently with feature points and object contours.

III. Notation

Consider a pin-hole camera model. Let $M = [X, Y, Z, 1]^T$ be a point (homogeneous coordinates) in the word-frame and $m = [u, v, 1]^T$ a point expressed in the CCD-frame. Under a full perspective projection it holds

$$sm = K\,[R\,|\,t]\,M \qquad \text{with} \quad K = \begin{pmatrix} a_u & c & u_0 \\ 0 & a_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

where s is an arbitrary scale factor, (R, t) are the rotation and translation of the camera frame with respect to the world frame, K is referred to as the camera intrinsic matrix, (u_0, v_0) are the coordinates of the principal point, a_u and a_v the scale factors in the image u and v axes, and c the parameter describing the skewness. We refer to *actual camera* (*desired*) as the camera frame in the actual (target or desired) position during the visual servoing. Vectors and matrices with suffix a (d) refers to the actual (desired) camera. Consider the main frame attached to the actual camera so that projection matrices on current and desired cameras become

$$P_a = K_a\,[I\,|\,0] \qquad P_d = K_d\,[R\,|\,t] \tag{1}$$

Hereafter, we assume that the two cameras have the same camera parameters $K_a = K_d = K$.

Finally, let us introduce the *bi-tangent* notation. Consider two views of the same scene and overlap the two

images. The term bi-tangent refers to a special line passing through the corresponding projections of the same $3D$ point or to the line that is tangent to two apparent contours of the same $3D$ object (Fig. 2).

IV. Epipolar Geometry

In this section we introduce the basic notation for the epipolar geometry and the mathematical formulation of the autoepipole configuration [9], [7], [12]. Consider a pair of cameras $\{a\}$ and $\{d\}$, with optical centers c_a and c_d and optical axes Z_a and Z_d, respectively. The segment $c_a c_d$ is called *baseline* and its intersections with the image planes define the *epipoles* e_a and e_d. A plane containing the baseline is referred to as the *epipolar plane*. The *epipolar line* is the intersection of an epipolar plane with the image plane. All the epipolar lines intersect at the epipole. Given a pair of views of a scene and a set of corresponding image points, m_{a_i} and m_{d_i}, there exists a matrix $F \in \mathcal{R}^{3\times 3}$ referred to as *fundamental matrix* [9], such that:

$$m_{a_i}{}^T F m_{d_i} = 0 \quad \forall i. \tag{2}$$

The fundamental matrix has rank 2 and is defined up to an arbitrary scale. For any point m_d (m_a) in one view, $l_a = F m_d$ ($l_d = F^T m_a$) defines the epipolar line in the other view such that the corresponding point m_a (m_d) belongs to this line. Moreover, the null right (one dimensional) space of F (F^T) represents the epipole e_d (e_a) on the image plane in homogeneous coordinates. The epipole can be interpreted as the projection on the image plane of the center of the other camera. The fundamental matrix depends on the camera parameters and relative position of the two cameras and can be expressed through the *essential matrix* E [15]

$$F = K^{-T} E K^{-1} \tag{3}$$

where E is defined as the product between the skew-symmetric matrix obtained from the translation vector t and the rotation matrix R between the two views $E = [t]_\times R$. The epipolar geometry is a powerful tool that can be also employed in problems dealing with apparent contours. Assume that the viewed scene consists of an object with a smooth surface, then the *contour generator* is defined as the locus of the points on the surface such that the tangent planes contain the camera center.

The contour generator divides the object surface into a visible part and an occluded one (for this reason it is also referred to as *occluding contour*). The perspective image of the contour generator on the image plane of the camera is referred to as *apparent contour*. An interesting property relates the apparent contours and the epipolar lines [22].

Property 1: Consider two views of the same scene. The epipolar lines, corresponding to the epipolar plane tangent to the object surface, are tangent to the apparent contours projected onto the image planes.

A. The epipolar geometry under pure translation

When the two camera positions undergo a pure translational displacement ($R = I$), the essential matrix and the fundamental matrix become

$$E = [t]_\times I = [t]_\times$$
$$F = [Kt]_\times K R K^{-1} = [Kt]_\times = [e]_\times. \tag{4}$$

The following property is a direct consequence of (4).

Property 2: Under pure translations, the two epipoles and the two epipolar lines are equal.

The correspondence of epipoles under pure translations is trivial since $\ker(F) = \ker(F^T) = e$. For the epipolar lines, consider two corresponding points m_d and m_a. From (2) we can define the actual epipolar line as

$$l_a = F m_d = [e]_\times m_d.$$

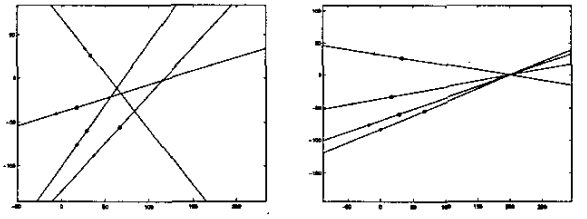
While the epipolar line in the desired image can be defined as the line passing thought m_d and the epipole e

$$l_d = e \times m_d = [e]_\times m_d.$$

It follows that the two epipolar lines have the same equations in the actual and desired images: $l_a = l_d$.

Property 2 is known as auto-epipolar property. In particular, if the camera motion is a pure translation, then the corresponding points are referred to as auto-epipolar. The auto-epipolar collinearity property does not hold for general motion [7], [12].

The auto-epipolar property holds also for apparent contours as shown in Fig. 2 where two different objects are viewed by two different viewpoints. The epipolar lines and the epipoles are simply computed by overlapping the two images, computing the bi-tangents to the contours and then their intersection.



(a) General roto-translation. (b) Pure translation.

Fig. 1. (a) Different and (b) equal orientation for the two views. In the first case all the bi-tangents obtained in the overlapped image do not intersect at a unique point. In the pure translation case all the bi-tangents intersect at the epipole. Crosses (circles) are the point projections on the actual (desired) camera.

V. Auto-Epipolar Visual Servoing

The main idea of the proposed visual servoing strategy is to exploit the auto-epipolar property to retrieve information about the relative orientation of two cameras. From the previous section the following statement holds true

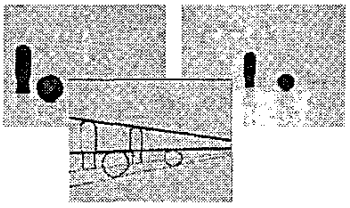


Fig. 2. Bi-tangents intersecting at the epipole for a scene with contours.

When the cameras have the same orientation all the bi-tangents intersect at the same point which is the epipole.

However, in order to exploit the auto-epipolar property, we need to prove that the same orientation is a necessary condition for having all the bi-tangents intersecting at the same point. This is because we want to use the one-point intersection condition as a control variable to compensate rotations in the visual servoing procedure.

We know that in general when a roto-translation occurs between the two cameras the bi-tangents will intersect at different points (Fig. 1). But can we guarantee that there exists no configurations for the two views such that all the bi-tangents intersect at one point even if the cameras undergo a non pure translation? In general this strongly depends on the number of feature points and on their relative pose in the 3D space.

Summarizing, given n corresponding feature points, the statement

$$R = I \;\Rightarrow \textit{ the } n \textit{ bi-tangents intersect at one point.}$$

holds true, while the statement

$$\textit{The } n \textit{ bi-tangents intersect at one point} \;\Rightarrow\; R = I \tag{5}$$

in general does not hold true. The following theorem, whose proof is given in [20], provides sufficient condition for statement (5) to hold true.

Theorem 1: Given 8 point matches m_{ai} and m_{di} in the two views and let matrix A built up as $A = [A_1, A_2, ..., A_8]^T$ with $A_i = m_{di} \otimes m_{ai}$ (the Kronecker product between m_{di} and m_{ai}). If matrix A has rank 8 then when all the bi-tangents intersect at the same point p a pure translation between the two views occurs ($R = I$) or $R = e^{[e_a]_\times \pi}$ and the point p is the epipole.

Note that in this work only planar motions of the cameras are considered so that only rotations around the Y-axis are allowed thus, under the assumption od the theorem, if all the bi-tangents intersect at the same point p, the relative orientation must be $R = I$.

A. The planar visual servoing algorithm

Consider a planar motion of the mobile robot with a camera mounted on and assume that all the three degrees of freedom of the camera-robot can be independently actuated (holonomic robot).

Assume that the feature points are such that the sufficient condition of Theorem 1 is satisfied. The main idea of the proposed visual servoing is to execute a rotation of the camera-robot such that all bi-tangents have a common intersecting point.

In Fig. 1 feature bi-tangents in the overlapped images are shown for two different relative positions of the current and desired cameras. Crosses (circles) are the point projections on the actual (desired) camera. In Fig. 1-a, the two cameras do not have the same orientation and the bi-tangents do not intersect at a single point. In Fig. 1-b only a translation occurs between the two cameras and all the bi-tangents pass through the epipole. Exploiting the auto-epipolar property shown in Fig. 1-b, it is possible to compensate the rotation between actual and target camera in the visual servoing strategy.

The main idea behind the algorithm consists in rotating the actual camera until all the bi-tangents intersect at the same point to get the same orientation between them. When only a translation is present, the intersection point is the *auto-epipole*. Once the rotation is compensated, only a translation occurs between the current and target camera positions. Several strategies can be used to translate the camera-robot to the target. For instance in [21], the author observes that the epipoles does not change when the robot moves along the baseline and uses this property to compensate translations. In this paper we propose to compensate the pure translation between the actual and desired camera-robot position executing two consecutive translations: the first along the Z-axis and the second along the X-axis of the camera-robot.

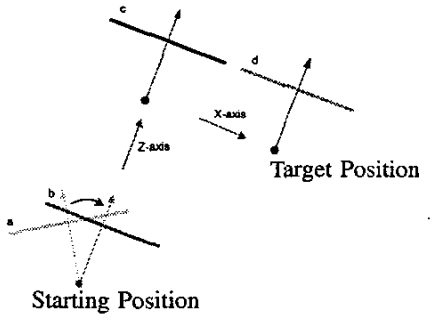


Fig. 3. Visual servoing trajectory and algorithm overview.

The typical trajectory of the camera-robot is reported in Fig. 3. The visual servoing is summarized as follows
Algorithm overview:

Phase 1: Rotate the actual camera until all the bi-tangents intersect at the same point (Fig. 3: from configuration a to b).

Phase 2: Get the coplanarity of the actual and target image

planes (Fig. 3: from configuration b to c). Note that this condition is equivalent to get a parallel condition between the baseline and the image planes.

Phase 3: Reach the target position translating along the X-axis (Fig. 3: from configuration c to d).

The visual servoing is decoupled in three phases. The first one compensates rotations while the other two compensate translations. All the camera parameters are supposed to be unknown. The proposed visual servoing can work indifferently with both feature points or contours (see Fig. 2) since both of them allow to define the bi-tangents used for visual servoing. This skill makes the proposed control strategy very flexible. In the next sections the single phases are analyzed.

B. Phase 1: compensating rotation

In this phase the visual servoing rotates the actual camera until the desired camera orientation is gained (Fig. 3: from configuration a to b). As discussed in Section IV-A, for general roto-translations between the two cameras, the bi-tangents passing through the overlapped features do not intersect at the same point, but when it happens, according to Theorem 1, the cameras are in auto-epipolar configuration and no relative orientation between them exists. A cost function approach is used to measure how far the actual camera is from the auto-epipolar configuration and to design the first phase of the visual servoing strategy. Let l_1, l_2 and l_3 be the bi-tangents of 3 feature points. In homogeneous coordinates the intersection point between two of them, e.g. l_2 and l_3 can be written as $p_{23} = l_2 \times l_3$. Moreover, equation $l_1 \cdot p_{23} = 0$ is satisfied when the point p_{23} belongs to the third bi-tangent l_1. Then the one-point intersection condition between the three bi-tangents can be written as

$$l_1 \cdot (l_2 \times l_3) = 0. \tag{6}$$

Equation (6) can be expressed in terms of determinants for bi-tangents l_i, l_j and l_k as

$$D_{ijk} := det \begin{bmatrix} l_i^T \\ l_j^T \\ l_k^T \end{bmatrix} = 0. \tag{7}$$

For more than 3 feature points, more than one determinant condition must be taken into account. It can be easily shown that in the case of 4 feature points two determinant conditions are needed, $D_{123} = 0$ and $D_{234} = 0$, to guarantee that all the bi-tangents intersect at the same point. Generalizing to n feature points, the constraints that must be satisfied to get the auto-epipolar property are

$$D_{(k-2)(k-1)k} = 0 \quad \text{for } k = 3, \ldots, n \tag{8}$$

The determinant condition (8) allows to build a cost function exploited for compensating the camera-robot rotations. The proposed cost function for n feature points is

$$\Psi(\theta) = \sum_{k=3}^{n} D_{(k-2)(k-1)k} \tag{9}$$

which depends upon the relative angle θ between the current and desired cameras and provides a measure on how far the auto-epipolar configuration is.

For a 5 feature points scene, the plot of determinants $D_{123}, D_{234}, D_{345}$ and the cost function Ψ (dashed line) are reported in Fig. 4 when the relative rotation goes from -40 to $+40$ degrees. The cost function is 0 when the current and target camera-robot positions undergo a pure translation (configuration b in Fig. 3).

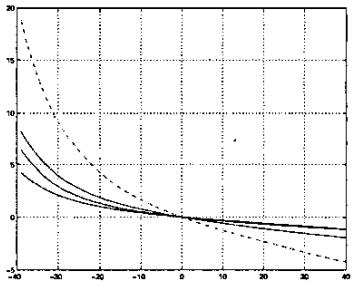


Fig. 4. Bi-tangents determinants (continuous lines) and overall cost function $\Psi(\theta)$ (dashed line).

The compensating rotation phase of the visual servoing can be simply designed relating the angular velocity of the camera-robot to the cost function (9). The simplest relationship is the proportional one ($\lambda > 0$)

$$\dot{\theta}(t) = \lambda \psi(\theta). \tag{10}$$

It can be shown that the controller is robust under large cameras' displacement ranges and in presence of measurements noise on the features. A further interesting property of this approach is that the rotation phase of the proposed visual servoing does not fail in singular configurations. The control law proposed in (10) works, for example, when the desired and current image planes are parallel to the baseline (the epipoles go to infinity) or when the two cameras undergo a pure rotation as well. Note that in the latter example the camera centers lay in the same position and the epipolar geometry is not well defined. During the visual servoing it may happen that some features go out from the field of view [6]. To overcome this problem a procedure to keep features in field of view have been introduced. This procedure simply "freezes" the rotation control when the features migrate out of the field of view and translates the camera along a suitable direction until all the features appear again in the field of view. Then the rotation control starts again. Further details can be found in [20].

C. Phase 2: the image planes are parallel to the baseline

In this section the motion from configuration b to c in Fig. 3 is described.

Refer to Fig. 3, any translation of the camera-robot along a direction which is not parallel to the image plane allows to reach the configuration c. The best translation is that along the optical axis which minimizes the length of the trajectory from configuration b to c. Throughout this phase, because only camera translations are involved, the auto-epipolar property holds. Despite the epipole's position changes due to the translations, the bi-tangents continue to intersect at the same point that is just the epipole. In this phase the goal, that has to be reached, is to gain the parallelism between the epipolar lines. This condition implies that the image planes are both parallel to the base-line and the epipoles go to infinity. Various control strategies can be designed in order to get the parallelism conditions [20].

D. Phase 3: reaching the target

At the end of Phase 2, the auto-epipole is at infinity, all the bi-tangents are parallel and the current and desired cameras exhibit only a relative translation along the X-axis parallel to the image planes and laying on the plane of the camera-robot (configurations c and d in Fig. 3).

A straightforward method to steer the current features on the target ones is to use a simple proportional controller on a a distance measurement between the feature centroids.

Although the two translation phases could be executed at the same time using a single control law, in practical cases we prefer to consider two consecutive steps. This choice was preferred to provide a more reliable and robust convergence to the goal.

VI. Real Experiments

The proposed visual servoing algorithm has been experimentally validated starting from many different initial poses and it efficiently converges. The experimental testbed consists of a CCD camera (HITACHI KP-D50) mounted on the holonomic mobile base Nomad XR4000 by Nomadic. The scene consists of 12 point features (the upper six points are reported in Fig. 5). Images, at 320×240 pixels, are acquired at 5Hz. A first filtering process is performed to solve the corresponding points problem. The computing system is an AMD Athlon at 1.4Ghz and the overall experiment took about 40secs. The real trajectory followed by the camera-robot during the experiment can be observed in Fig. 5 (both position and orientations are shown). After the acquisition of the target image the camera-robot was moved $1.5m$ far from the target position and the orientation, between the starting and the target cameras, was set to ± 30deg and ± 20deg in four different trials. The initial and target camera-robot

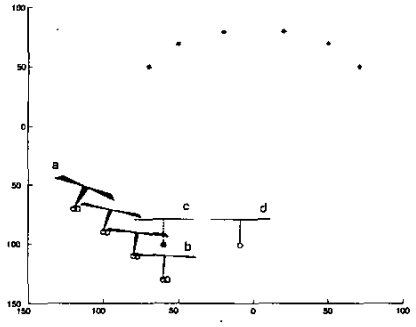


Fig. 5. Visual servoing trajectory.

positions are reported in Fig. 5 and labelled with a and d, respectively.

Phase 1: During this phase the camera-robot moves from configuration a to configuration b in Fig. 5. The motion is not a pure rotation because the procedure for keeping the features in the field of view was working properly. The relative orientation errors for several starting configurations are shown in Fig. 6. For these trials the determinant based cost function (9) is reported in Fig. 7. During Phase 1, the procedure for keeping the features in the field of view stops the rotation and actuates a translation until the features are visible again. The steady state error on the rotation angle is about ± 2deg.

Phase2: In this phase, the camera-robot moves from configuration b to configuration c in Fig. 5. The controlled variable is the second singular value of a matrix built with bi-tangent lines which appeared to be robust with respect to the measurement noise [20]. In Fig. 8 the distance from the configuration c is reported. The distance are in *cm* and the steady state error is due to the proportional nature of the control law.

Phase 3: The last phase is accomplished with good performances also in presence of the steady state errors given by the previous steps.

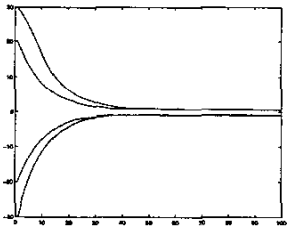


Fig. 6. Rotation angle errors during the visual servoing (Phase 1). Different experiments are reported which starts from different starting points.

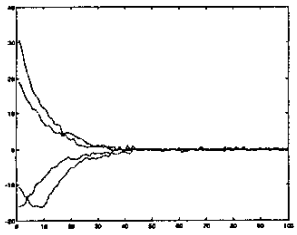


Fig. 7. The determinant based cost function $\Psi(\theta)$ during Phase 1.

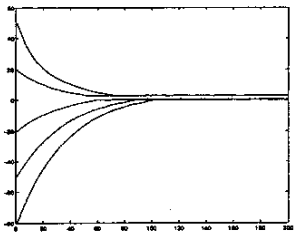


Fig. 8. Distance from configuration c for several trials (Phase 2).

VII. Conclusions

The auto-epipolar properties of the feature points under camera pure translation motion were exploited to design a new image-based visual servoing algorithm. The main idea underlying the auto-epipolar visual servoing consists in compensating the orientation between the current and desired camera positions and then executing two consecutive translations along the Z-axis and X-axis to reach the final target. The algorithm does not need the knowledge of the internal camera parameters and works indifferently with both corresponding points and contours. In the present version, the method requires that camera-robot moves on a plane but the optical axis can be oriented in any direction with respect to the floor plane. Experiments have been executed to validate the visual servoing approach. Experimental results prove that the overall process is robust to measurement noises. Work is in progress to generalize the visual servoing based on the auto-epipoles to non planar motions.

VIII. References

[1] R. Basri, E. Rivlin, and I. Shimshoni. Visual homing: Surfing on the epipoles. In *ICCV*, pages 863–869, 1998.

[2] Z. Bien, W. Jang, and J. Park. Characterization and use of feature jacobian matrix for visual servoing. pages 317–363, 1993.

[3] G. Chesi, E. Malis, and R. Cipolla. Automatic segmentation and matching of planar contours for visual servoing. In *International Conference Robotics and Automation, ICRA '00.*, volume 3, pages 2753 –2758, 2000.

[4] G. Chesi, J. Piazzi, D. Prattichizzo, and A.Vicino. Epipole-based visual servoing using profiles. In *IFAC'02 World Congress*, Barcellona. Spain., July 2002.

[5] G. Chesi, D. Prattichizzo, and A. Vicino. A visual servoing algorithm based on epipolar geometry. In *IEEE Int. Conf. Robotics and Automation*, May 2001.

[6] G. Chesi, D. Prattichizzo, A. Vicino, and K. Hashimoto. A visual servoing strategy for keeping features in the field's of view. In *Proc. IEEE Int. Conf. on Robotics and Automation*, In press 2003.

[7] R. Cipolla and P.J. Giblin. *Visual Motion of Curves and Surfaces*. Cambridge University Press, 2000.

[8] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. In *IEEE Trans. on Robotics and Automation*, volume 8(3), 313–326 1992.

[9] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, 1993.

[10] G. D. Hager. A modular system for positioning using feedback from stereo vision. In *IEEE Trans Rob and Auto*, volume 13(4), pages 582–595, 1997.

[11] G.D. Hager, G. Grrunwald, and K. Toyama. Feature-based visual servoing and its application to telerobotics. *In Intelligent Robots and Systems, Elsevier*, pages 72–94, 1995.

[12] R. Hartley and A. Zisserman. *Multiple view in computer vision*. Cambridge University Press, September 2000.

[13] S. A. Hutchinson, G. D. Hager, and P. I. Corke. A tutorial on visual servo control. *IEEE Trans. Robotics and Automation*, 12(5):651–670, citeseer.nj.nec.com/hutchinson96tutorial.html 1996.

[14] K.Hashimoto and T.Noritsugu. Performance and sensitivity in visual servoing. In *IEEE Int. Conf. on Robotics and Automation*, pages 384–390, Leuven, Belgium, May 1998.

[15] H.C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.

[16] E. Malis, F. Chaumette, and S. Boudet. 2d 1/2 visual servoing stability analysis with respect to camera calibration errors. In *Proceedings of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems*, page 691, 1998.

[17] A. Marotta, J. Piazzi, D. Prattichizzo, and A. Vicino. Epipole-based 3d visual servoing. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Sep. 2002.

[18] P.R.S. Mendonça and R. Cipolla. Estimation of epipolar geometry from apparent contours: affine and circular motion cases. In *Computer Vision and Pattern Recognition*, volume 1, page 1009, June 1999.

[19] P.R.S. Mendonça, K.Y.K. Wong, and R. Cipolla. Camera pose estimation and reconstruction from image profiles under circular motion. In D. Vernon, editor, *Proc. 6th European Conference on Computer Vision*, volume II, pages 864–877, Dublin, Ireland, June 2000. Springer–Verlag.

[20] J. Piazzi and D. Prattichizzo. An auto-epipolar strategy for mobile robot visual servoing. Technical report, University of Siena, 2003.

[21] P. Rives. Visual servoing based on epipolar geometry. In *International Conference on Intelligent Robots and Systems*, volume 1, pages 602–607, 2000.

[22] K. Åström and F. Kahl. Motion estimation in image sequence using the deformation of apparent contours. In *IEEE Transactions on Pattern Analysis and Machine Intelligence, 21(2)*, 1999.

[23] K. Åström R. Cipolla and P.J. Giblin. Generalised epipolar constraints. In *International Journal of Computer Vision*, number 2, pages 97–108, 1996.

[24] C.J. Taylor, J.P. Ostrowski, and S.H. Jung. Robust visual servoing based on relative orientation. In *IEEE Conf. on Comp. Vision and Patt. Recog*, pages 574–580, June 1999.