Auto-epipolar Visual Servoing

Jacopo Piazzi
Johns Hopkins University
Baltimore, Maryland USA

Domenico Prattichizzo
Univeristá degli Studi di Siena
Siena, Italy

Noah J. Cowan
Johns Hopkins University
Baltimore, Maryland USA

***Abstract*— We present a purely rotational visual servoing algorithm which aligns the orientation between two cameras at different locations in space. Specifically, our kinematic controller steers a set of so-called *bi-tangent lines* to intersect at the epipole using a purely image-based bi-tangent line Jacobian. Bi-tangent lines, i.e. lines joining corresponding features on the superposition of two views of a scene, can be defined for both points and contours, so we apply our controller to both feature types. Simulated experiments demonstrate the parametric robustness of the proposed method.**

I. Introduction

Achieving rotational alignment between two cameras simplifies certain problems in vision, such as scene reconstruction [1] and camera calibration [2]. In this paper we present a visual servoing (VS) algorithm that rotationally aligns two cameras by steering a set of bi-tangent lines, i.e. the lines joining the corresponding features on two superimposed views, to intersect at a common point – the epipole. The co-intersection of at least eight bi-tangents ensures that the essential matrix between the two cameras is skew-symmetric, thus ensuring alignment of the cameras (modulo a 180^o ambiguity in rotation about the epipole). The rotational error between the current and desired camera converges *independent of the translational displacement*; in this sense, the controller presented decouples the rotational and translational degrees of freedom.

Zisserman and Mundy [3] introduced the idea of bi-tangents which Cipolla and Sato [4] exploited to recover epipoles and the fundamental matrix under pure camera translation; the rotational control algorithm in this paper builds directly on these ideas. Prior works on VS also employ epipolar geometry: Rives describes an algorithm that drives robot to a desired pose by steering the image features along epipolar lines [5]; Basri and Shimshoni recover the rotation matrix from the decomposition of the essential matrix, and they retrieve the translation direction from the epipoles [6]. Similarly Malis *et al.* [7] and Taylor *et al.* [8], decompose the homography or essential matrices, respectively, to recover the rotation between two views. These methods decouple rotational and translational control by explicitly estimating the rotation matrix from the essential or homography matrices. In our method we do not require the decomposition of any matrix related to the camera geometry. To the best of our knowledge, no other purely image-based VS strategies to date can rotationally align two cameras.

Our controller aligns the orientation between two cameras independent of the translational error without recovering either the rotation matrix between the cameras or the feature depths. Our method drives a set of bi-tangents to co-intersect via an image-based objective function, the gradient of which is projected through the transposed or pseudo-inverse of a *purely image-based* Jacobian matrix. Thus, the features and resulting controller are easy to compute from image measurements. Since bi-tangents are well defined for points or contours, our approach enables the control of a camera's motion when the scene presents either feature type. Contour features render point tracking (much less point correspondence between views) challenging or impossible which may explain why epipolar geometry estimation from contours has been limited to special settings [9]–[11]. To apply our controller in the context of contour features, we first present a method to estimate the epipolar geometry from corresponding contours, and use the result to align two cameras to the same orientation.

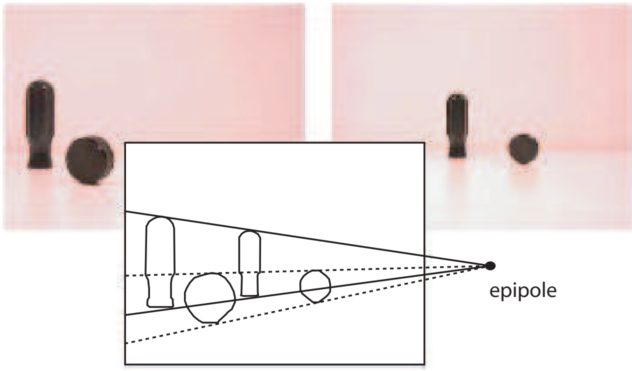

Fig. 1. Bi-tangents to contours in a superimposed image. When the cameras undergo pure translation, the bi-tangents intersect at the epipole, as shown.

II. Bi-tangents for Points and Contours

Consider a fully calibrated perspective projection camera and a point $p \in \mathbb{R}^3$, expressed in a camera-fixed frame, viewed by the camera. Without loss of generality, attach the camera frame to the optical center and align the Z-axis along the optical axis. Remap the image-plane to a sphere to recover the symmetry that is "broken" by a flat image plane [12], thus each image measurement defines a unit vector

$$y = \pi(p) := \frac{p}{\|p\|}.$$

During a training phase, we command the camera to move a desired location and acquire an image, and store the projections y_{d_i}, $i = 1, \ldots, N$, of several points in the scene. At each time instant during VS, we measure the location of these points in the current (or "actual") view, y_{a_i}, $i = 1, \ldots, N$.

Consider the image obtained by superimposing two views of the same scene. For each pair of corresponding features define the line tangent to both as the *bi-tangent*. Compute the bi-tangent from a pair of corresponding points y_a, y_d as

$$b \sim \hat{y}_a y_d \tag{1}$$

where $\sim$ denotes equality up to a scale and $\hat{x}$ denotes the usual skew symmetric matrix obtained from the vector $x \in \mathbb{R}^3$.

Given an object in space, the *apparent contour* is the projection of the *contour generator*, i.e. the locus of points on the object generated by tangent planes through the optical center [13]. Consider the image obtained by superimposing two views of the same contour (they will, in general, be images of different contour generators). A line tangent to both curves such that both curves lie in the same half plane defined by the line is a *contour bi-tangent*. Each pair of corresponding contours may generate two bi-tangents as shown in Fig. 1.

III. AUTO-EPIPOLAR GEOMETRY

We now review facts from the epipolar geometry [14], [15] and the mathematical formulation of the auto-epipolar property [3], [4], which states that under pure translation, bi-tangents intersect at the epipoles. We then present new results relating bi-tangents to the relative orientation between two views.

A. Review of Epipolar Geometry

Consider the *actual* and *desired* cameras with optical centers c_a and c_d and optical axes Z_a and Z_d, respectively. The segment $c_a c_d$, called the *baseline*, intersects with the image planes at the two *epipoles* e_a and e_d. A plane containing the baseline, called an *epipolar plane*, intersects the image at an *epipolar line*. All the epipolar lines intersect at the epipole.

Given a pair of cameras and their views of a scene there exists a matrix $E \in \mathbb{R}^{3\times3}$ called the *essential matrix* [14], such that

$${y_{a_i}}^T E y_{d_i} = 0 \tag{2}$$

for all corresponding image points, y_{a_i} and y_{d_i}, $i = 1, \ldots, N$. In general the essential matrix has rank 2 and is defined up to an arbitrary scale. For any point y_d in one view, $\ell_a = E y_d$ defines the epipolar line in the other view such that the corresponding point y_a belongs to this line. Likewise, given any point y_a, the corresponding epipolar line in the other view is given by $\ell_d = E^T y_a$. Since all epipolar lines co-intersect, the (one dimensional) right null space of E contains the epipole e_d, i.e. $\ker(E) = \operatorname{span}\{e_d\}$. Similarly $\ker(E^T) = \operatorname{span}\{e_a\}$.

The essential matrix depends on the relative position and orientation of cameras, namely

$$E = \hat{T} R, \tag{3}$$

where $\hat{T}$ is the skew-symmetric matrix obtained from the relative translation vector T and R is the rotation matrix between the actual and desired cameras.

Given two camera views of the same smooth object, their contour generators intersect at the common visible a point, called a *frontier point* [16], [17]. An epipolar plane is tangent to a surface at the frontier point, and the epipolar line in each views is tangent to the apparent contour at the projection of the frontier point.

B. The auto-epipolar property

In general, an essential matrix takes the form of (3), but certain configurations, called *auto-epipolar configurations*, cause the essential matrix to be skew symmetric. For example, under pure translational displacement between two cameras (i.e. $R = I$), we have that

$$E = \hat{T} I = \hat{T}.$$

Also note that the corresponding epipolar lines and epipoles, referred to as *auto-epipoles*, are equal [3], [4].

Proposition 1 (Auto-epipolar property): When the essential matrix is skew symmetric the two epipoles and the epipolar lines in both views are equal.

Proof: The correspondence of epipoles results from the fact that $\ker(E) = \ker(E^T) = \operatorname{span}\{e\}$, where $e = e_a = e_d$. Note that the $T \sim e$ and the essential matrix can be expressed as

$$E = \hat{e}. \tag{4}$$

To see the equality of epipolar lines for a corresponding pair of points, note that the epipolar line in the actual view is given by $\ell_a \sim E y_d$. Moreover the epipolar line in the desired image can be defined as the line passing thought y_d and the epipole e, namely

$$\ell_d \sim e \times y_d = \hat{e} y_d = E y_d \sim \ell_a.$$

■

Although a pure translational displacement ensures a skew symmetric essential matrix, there are other auto-epipolar configurations as well.

Lemma 1 (Ma et al. [18]): Let $T \in \mathbb{R}^3$ and $R \in \mathrm{SO}(3)$. If $\hat{T}R$ is skew symmetric matrix, then $R = I$ or $R = e^{\hat{u}\pi}$ where $u = \frac{T}{||T||}$. Further, $\hat{T} e^{\hat{u}\pi} = -\hat{T}$.

The $e^{\hat{u}\theta}$ denotes the exponential representation of a rotation matrix. Lemma 1 describes the two cases for which the essential matrix is skew symmetric: pure translation and relative rotation around the baseline of 180 degrees.

While the epipolar lines have a physical interpretation, i.e. they can be considered as the intersection between the epipolar plane and the image plane, the physical interpretation of the bi-tangents is less obvious. The following proposition captures the relationship between bi-tangents and epipolar lines when the essential matrix is skew-symmetric.

Proposition 2: If the essential matrix is skew-symmetric then a bi-tangent line between corresponding features $b_i \sim \hat{y}_{a_i} y_{d_i}$ coincides with the epipolar line $\ell_i \sim \ell_{a_i} \sim \ell_{d_i}$.

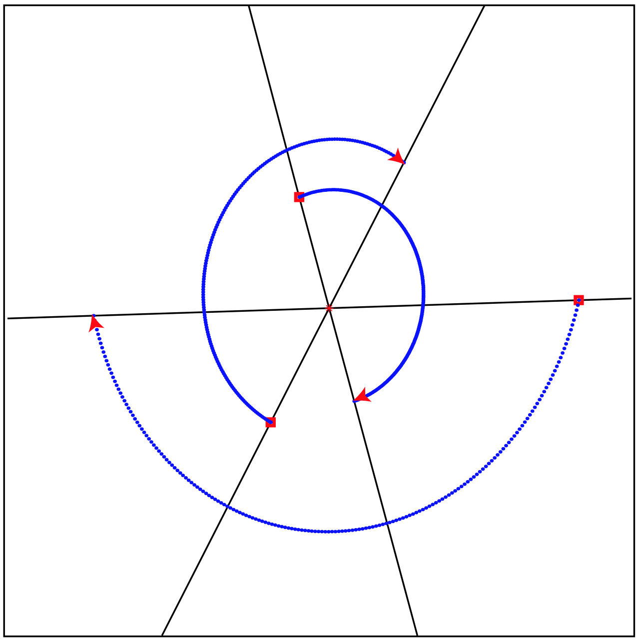

Fig. 2. Three features points and their respective epipolar lines. If the cameras are initially separated by a only a translation and then one camera rotates around the baseline by 180^o, then the features move as shown. Note that at the end of the rotation the features lie on the original epipolar lines.

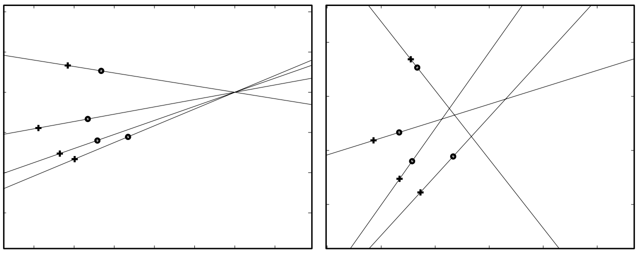

Fig. 3. Bi-tangent lines. *Left.* A set of points in the actual image (crosses), their corresponding locations in the desired image (circles) and the associated epipolar lines in auto-epipolar configuration. In this case the epipolar lines coincide with the bi-tangent lines in the super-imposed image. *Right.* In general configuration, the bi-tangents do not intersect at the epipole nor at any other special point, nor are they directly related to the epipolar lines.

Proof: By Proposition 1, $E = -E^T$. This implies that the epipolar lines are equal in both the images and thus they can be computed as the lines joining the actual and desired points; this is just the definition of the bi-tangent line. ■

Cipolla and Sato exploited this idea to retrieve the epipolar geometry under the pure translation between two cameras [4]. Under pure translation, the essential matrix is skew symmetric, and thus epipolar lines coincide with their corresponding bi-tangents. The intersection of at least two bi-tangents enables the construction of the epipole and subsequently the essential matrix from (4).

If system is in the auto-epipolar configuration, then all bi-tangents intersect at the same point, the epipole, as shown in Fig. 3. The converse strongly depends on the number of feature points considered and on their relative pose in the 3D space.

Theorem 1 (Piazzi and Prattichizzo [19]): Consider 8 corresponding points matches $y_{1(i)}$ and $y_{2(i)}$ in two respective views, and let the matrix $A = [a_1, a_2, ..., a_8]^T$ with[1] $a_i = (y_{2(i)} \otimes y_{1(i)})^T$. If matrix $A \in \mathbb{R}^{9\times 8}$ has rank 8 then when all the bi-tangents intersect at the same point $\mathbf{p}$, the essential matrix is skew, and the point $\mathbf{p}$ is the epipole.

Note that when 3D points are in general configuration, i.e. they do not belong to any critical surfaces (like a plane) [20], the rank of A is always eight[2].

C. Epipolar Geometry from contours

Theorem 1 allows one to determine if the essential matrix is skew symmetric without estimating the epipolar geometry, given a set of corresponding points. We suspect the same result applies to contours.

Conjecture 1: Given $2n \geq 8$ bi-tangents (2 for each contour) obtained by a set of n corresponding contours $C_{a(obj)i}$ and $C_{d(obj)i}$ induced by spherical objects on the scene. If the $2n$ bi-tangents intersect at the same point then the cameras are in auto-epipolar configuration.

Note that contour bi-tangents intersect the corresponding contours at points that *do not correspond*. Therefore we cannot apply Theorem 1. Thus far, the controller simulations presented in the next section have not converged to a configuration that disproves our conjecture, but the proof represents work in progress.

One way to determine if a set of bi-tangents intersect at the same point is to observe the smallest singular value of the matrix containing all the lines [19]. If this value is equal to zero then the bi-tangents have a single intersection point. The following corollary, based on Conjecture 1, formalizes this fact.

Corollary 1: Consider two cameras in a generic rigid body motion with relative orientation R and $2n \geq 8$ bi-tangents (2 for each contour) obtained by a set of n corresponding contours $C_{a(obj)i}$ and $C_{d(obj)i}$ induced by spherical objects on the scene. Given the matrix

$$M_{R_x} = \begin{pmatrix} b_{R_x1} \\ b_{R_x2} \\ \vdots \\ b_{R_xn} \end{pmatrix}, \quad (5)$$

where a b_{R_xi} is a bi-tangent obtained by two corresponding contours $C_{d(obj)i}$ and $R_x C_{a(obj)i}$ (by which we mean that every point of the contour is multiplied by the rotation matrix R_x). The smallest singular value of M_{R_x} is equal to zero if and only if $R_x = R^T$ or $R_x = e^{\hat{u}\pi}R^T$ where $u = \frac{ep_d}{||ep_d||}$.

Proof: Suppose we are given the contours $C_{a(obj)1}$ and $C_{d(obj)1}$ induced by the same sphere, multiplying the points of the $C_{a(obj)1}$ by the matrix R^T or $e^{\hat{u}\pi}R^T$, they are oriented in points of a contour that is in auto-epipolar

[1] The symbol $\otimes$ is the Kronecker product and the resulting vector $a_i \in \mathbb{R}^{9\times 1}$

[2] The theorem hypothesis are the same of the famous 8-points algorithm. Therefore if the epipolar geometry can be estimated from the points then the theorem holds true

configuration with the corresponding contour $C_{d(obj)1}$. Since every contour $C_{a(obj)i}$ is "moved" by the matrix R in auto-epipolar configuration with the corresponding contour $C_{d(obj)i}$ the bi-tangents intersect at the same point e_d. The matrix M_{R_x} is thus singular, because the intersection point e_d represents the right null vector of M_{R_x} and the smallest singular value is zero. On the other hand if the smallest singular value of M_{R_x} is zero, the bi-tangents have a single intersection point. But from Conjecture 1 the cameras are in auto-epipolar configuration, and the only two possible rotation matrices to obtain such a configuration are $R_x = R^T$ or $R_x = e^{\hat{u}\pi} R^T$, i.e. the relative orientation matrix between the camera premultiplied by I or the rotation around the base line $e^{\hat{u}\pi}$. ■

IV. 3D ROTATIONAL CONTROLLER

A. Bi-tangent Line Jacobian (BLJ)

We propose a rotational controller that steers all the bi-tangents to intersect at the desired epipole e_d. Our controller requires a bi-tangent line Jacobian matrix that relates the bi-tangent line velocities to the camera angular velocity, computed as follows.

Assume pure rotational motion of the actual camera. Given an image point, y_a, its velocity is given by

$$\dot{y} = \hat{y}_a \omega, \tag{6}$$

where ω is the body-frame representation of the camera angular velocity. From (1), the cross product of two views of the point

$$m = y_d \times y_a = \hat{y}_d y_a, \tag{7}$$

define a bi-tangent up to a scale, namely $b = km$, $k \neq 0$. Without loss of generality, we normalize the bi-tangent:

$$b = \frac{m}{\|m\|}. \tag{8}$$

Since the angular velocity is applied only to the actual camera, the point on the desired views y_d remains constant. Thus differentiating (8) yields:

$$\dot{b} = \Pi \, \hat{y}_d \dot{y}_a, \quad \text{where} \quad \Pi = \frac{1}{\|m\|}\Big(I - \frac{m^T m}{\|m\|\|m\|}\Big). \tag{9}$$

From (6), we have

$$\dot{b} = \Pi \, \hat{y}_d \hat{y}_a \omega = B\omega, \tag{10}$$

where the matrix $B = \Pi \hat{y}_d y_a \in \mathbb{R}^{3\times 3}$ is the rotational, image-based bi-tangent line Jacobian (BLJ).

B. Controller design

Our controller attempts to steer all bi-tangents to intersect at the desired epipole, which remains stationary during the process. Recall that the epipole is defined as the intersection between the baseline and the image plane, and since the actual camera rotates, but does not translate, the baseline does not change throughout the control procedure.

Define the image error as the distance between the bi-tangents and the epipole, $d_i = b_i^T e_d$, which vanishes only when the point e_d belongs to the line b_i. Since the epipole, e_d, remains constant, we have that

$$\dot{d} = J\,\omega \tag{11}$$

where $d = [d_1, d_2, \ldots, d_n]^T$,

$$J = (I_{n\times n} \otimes e_d^T) \begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_n \end{pmatrix}$$

and B_i is the BLJ associated with the i-th bi-tangent line. A rotational control strategy that (locally) minimizes the error d is given by

$$\omega = -\lambda J^{\dagger} d \tag{12}$$

where λ is a positive scalar gain and $J^{\dagger} = (J^T J)^{-1} J^T$ is the pseudo inverse of J. The simple gradient based approach for the input angular velocity (12), guarantees the convergence of the VS task in a neighborhood of the target, not unlike most the classical VS methods.

V. SIMULATION EXPERIMENTS

The control law (12) steers all the bi-tangents to intersect at the desired epipole in order to align the orientation between the two views. Since both points and contours induce bi-tangents, we report simulation results for both feature types. Our approach assumes no prior knowledge about the 3D scene, and assume that the internal camera parameters are known. We make no effort in the present work to maintain features within the FOV, which is left for future work [21].

A. Point feature experiments

Given a set of points in space in a general configuration we first estimate the desired epipole by computing the essential matrix through the linear 8-point algorithm [22]. We assume the ideal case of noiseless image points, from which the bi-tangent lines and the BLJ are computed at each sampling step. Fig. 5 (*Left*) shows an experiment with an initial orientation of $[-40°, -10°, -50°]$ (roll, pitch, yaw) and translation of $[70, -20, -10]$ (cm), relative to the desired camera. The control law in (12) leads to exponential decay in the error.

Although the control law presented in (12) guarantees local exponential convergence, it requires an estimate of the epipole, which may be corrupted by noisy image data, consequently degrading the performance of the controller. A planar auto-epipolar VS controller, presented in [19], does not require the estimation of the epipolar geometry; however extension of that simple "epipole-free" control law to the more general 3D case represents work in progress.

Another issue involves the two possible orientations assumed by the actual camera in the auto-epipolar configuration. According to Lemma 1, there are two global minima for the error, $\|d\|$: equal orientation, and a relative rotation around the baseline of 180^o. The problem is similar to the reconstruction of the camera pose from the decomposition of the essential matrix, which in general yields four possible solutions. Consider however that when the cameras are aligned to the same orientation, any translation maintains the auto-epipolar configuration, i.e. the bi-tangents continue to intersect at the same point. This does not hold for the other auto-epipolar configuration. In this case, any translation (except along the baseline) causes the bi-tangents to diverge from co-intersection. In this case we execute a rotation around the baseline of 180 degrees in order to align the cameras.

B. Contour feature experiments

We propose a method to estimate the epipolar geometry and align the orientation of two cameras when the scene exhibits only smooth objects. In general, corresponding points cannot be determined from corresponding contours. This complicates the estimation of the epipolar geometry because we can not apply the same procedures as for point features (e.g. the 8-point algorithm). However the curves obtained from apparent contours permit us to define bi-tangent lines from which we estimate the relative rotation matrix and the desired epipole as follows.

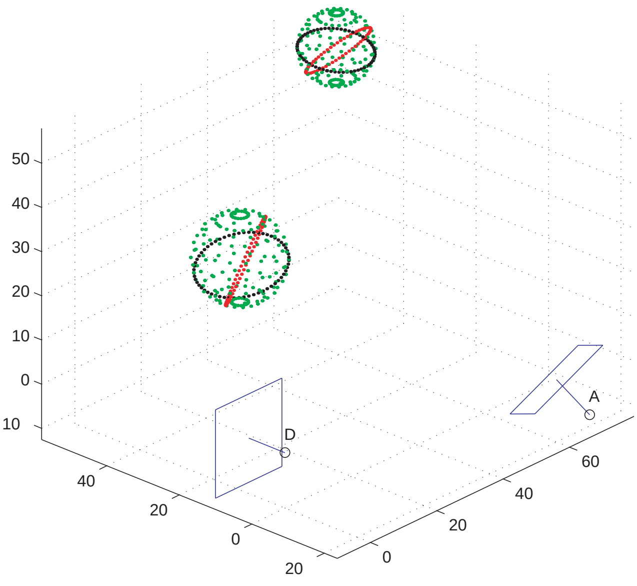

Fig. 4. Two spherical features and their contour generators.

Fig. 4 shows a scene with 2 spheres in general displacement with the contour generators induced by the cameras.

We need only estimate the epipolar geometry in the preliminary step,

According to Conjecture 1, M_{R_x} loses rank only when $R_x = R^T$ or $R_x = e^{\hat{u}\pi} R^T$. Thus we can estimate one of them by calculating the smallest singular value of M_{R_x}. We perform an optimization process in three variables to retrieve the unit vector axis and the angle of rotation, i.e. through the Rodrigues' formula, employing the smallest singular value of the matrix M_R as a scalar valued cost function.

Recovering relative rotation, enables the recovery of the entire epipolar geometry. The desired epipole is the intersection point of the bi-tangents and the rotation matrix is given by the optimization process. Therefore we build the essential matrix according to (3). Note that whatever R is obtained from the optimization process ($R_x = R^T$ or $R_x = e^{\hat{u}\pi} R^T$), the resulting essential matrix is the same (up to a scale).

Note that we need only estimate e_d at the very beginning of the processes – not at each time step. Naive optimization in Matlab (using fminsearch) requires approximately 10 seconds on a Pentium II, 650MHz. (This need only be executed once at the beginning of the VS process.) Note that our approach to estimating the epipolar geometry requires only three parameters (compared, for example, to the eight required for estimating the minimal parameterization of the fundamental matrix presented in [23]), and thus we believe a more efficient method can be found.

Once the epipolar geometry has been recovered we apply the same procedure as used previously for the points features. At every step of the process we compute the bi-tangents from the edges of the spheres and we employ the tangent points to build up the BLJ according to (10).[3]

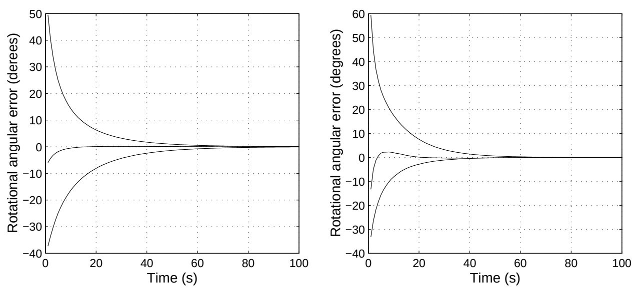

Fig. 5. Rotational error (using Euler angles) between camera frames during bi-tangent servoing. *Left.* Feature point based visual servoing. *Right.* Contour based visual servoing.

We ran many simulations starting from several initial misalignments. Fig. 5 (*Right*) shows the profiles of the angular errors between the two views with an initial orientation of about $[60°, -12°, -34°]$ (roll, pitch, yaw). In this case, despite large initial misalignment, the error converges to zero.

VI. Conclusion and Future Work

We presented a method to retrieve the epipolar geometry and to align the orientation of two views by exploiting bi-tangent line features. The procedure applies to both contours and feature points. In addition, we propose a procedure to estimate of the epipolar geometry for contours. One of the key steps involves computing the bi-tangent line Jacobian purely from image measurements.

This work takes a preliminary look at "auto-epipolar" geometry, but there remain several open questions. We

[3]Note that the BLJ in (10) is only approximate, since it is based on lines induced from corresponding points. In the presence of contours the bi-tangent line velocities depend in a complex way on the change of contours in the image. However, as confirmed by the experiment presented in Fig. 5 (*Right*), the BLJ provides good convergence. We suspect that the variation of the line tangent point on a smooth contour can be approximated locally as the variation of single point.

wish to relax the requirement of computing the epipoles to enable a VS implementation completely free of epipolar geometry estimation as done in [19] for the planar case. The field of view is another open issue: note that transients may cause the loss of features from the image. Finally, using contours instead of points may generate occlusions between the features that may imperil the computation of the bi-tangents.

This paper provides insight into the use of epipolar geometry to visually control a robot's motion. We believe that a fast and robust way to control the orientation could leave many open doors for solving VS problems. The aim of this work is to show that it is possible to design a method for aligning the camera that is, in principle, independent from the other degrees of freedom. In [24] we presented a completely decoupled translational and rotational controller with a large domain of attraction that respects the field of view. Our research line is to split the entire VS process in several sub-tasks and tackle each of them with robust and reliable control laws, with well characterized domains of attraction. It is our belief that such controllers can be stitched together [25] to create a globally convergent vision-based navigation system.

REFERENCES

[1] T. Moons, L. V. Gool, M. Proesman, and E. Pauwels, "Affine reconstruction from perspective image pairs with a relative object-camera translation in between," *Pattern Analysis and Machine Intelligence*, vol. 18, no. 1, 1996.

[2] J. Jang and K. Hong, "Self-calibration of a stereo-camera by pure translational motion," *International conference on Image Processing*, vol. 1, pp. 297–300, 1996.

[3] A. Zisserman and J. L. Mundy, "Repeated structures: Image corrispondence constraints and 3d structure recovery," *In J. L. Mundy, A. Zisserman and D. A. Forsyth editors "Applications of Invariants in Computer Vision"*, vol. (LNCS 825) Springer-Verlag, pp. 89–106, 1994.

[4] J. Sato and R. Cipolla, "Affine reconstruction of curved surfaces from uncalibrated views of apparent contours." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 11, pp. 1186–1198, 1999.

[5] P. Rives, "Visual servoing based on epipolar geometry," in *International Conference on Intelligent Robots and Systems*, vol. 1, 2000, pp. 602–607.

[6] R. Basri, E. Rivlin, and I. Shimshoni, "Visual homing: Surfing on the epipoles." in *ICCV*, 1998, pp. 863–869.

[7] E. Malis, F. Chaumette, and S. Boudet, "2d 1/2 visual servoing," *IEEE Transaction on Robotics and Automation*, vol. 15, pp. 328–250, Apr. 1999.

[8] 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*, June 1999, pp. 574–580.

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

[10] R. Cipolla and J. Sato, "Affine reconstruction of curved surfaces from uncalibrated views of apparent contours." *ICCV*, pp. 715–720, 1998.

[11] K.-Y. K. Wong, P. R. S. Mendonça, and R. Cipolla, "Reconstruction and motion estimation from apparent contours under circular motion," in *Proc. British Machine Vision Conference 1999*, T. Pridmore and D. Elliman, Eds., vol. 1. Nottingham, UK: British Machine Vision Association, September 1999, pp. 83–92.

[12] T. Hamel and R. Mahony, "Visual servong of an under-actuated dynamic rigid-body system: An image-based approach." *IEEE Transactions on Robotics and Automation*, vol. 18(2), p. 187198, April 2002.

[13] K. A. ström, R. Cipolla, and P. Giblin, "Motion from the frontier of curved surfaces." in *Fifth International Conference on Computer Vision*, Boston, USA, 1995.

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

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

[16] P. J. Giblin and R. Weiss, "Epipolar fileds on surfaces," in *Third European Conf. Computer Vision*, vol. 1, Stockholm, May 1994, pp. 14–23.

[17] J. H. Rieger, "Three dimensional motion from fixed points of a deforming profile curve," *Optics Letters*, vol. 11, pp. 123–125, 1896.

[18] Y. Ma, J. Košecká, and S. S. Sastry, "Linear differential algorithm for motion recovery: A geometric approach," *Internation Journal of Computer Vision*, vol. 36(1), pp. 71–89, Sep. 2000.

[19] J. Piazzi and D. Prattichizzo, "An auto-epipolar strategy for mobile robot visual servoing," in *International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas (USA), Sep. 2003.

[20] S. J. Maybank, "The projective geometry of anbiguous surfaces," *Philosofical Transaction of the Royal Society of London*, vol. A 332:1 - 47, 1990.

[21] N. J. Cowan, J. D. Weingarten, and D. E. Koditschek, "Visual servoing via navigation functions," *Transactions on Robotics and Automation*, vol. 18, no. 4, 2002.

[22] R. Hartley, "In defence of the 8-point algorithm." in *Proc. of IEEE Int. Conference on Computer Vision*, Cambridge MA,USA, June 1995, pp. 1064–1070.

[23] P. R. S. Mendonça and R. Cipolla, "Estimation of epipolar geometry from apparent contours: affine and circular motion cases." *Computer Vision and Pattern Recognition*, vol. 1, June 1999.

[24] J. Piazzi and N. J. Cowan, "Multi-view visual servoing using epipoles," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sendai. Japan, September 2004.

[25] R. R. Burridge, A. A. Rizzi, and D. E. Koditscheck, "Sequential composition of dynamically dexterous robot behaviors." *Int. J. Rob. Res.*, vol. 18(6), pp. 534–555, 1999.