Combining Cartesian and cylindrical coordinates in IBVS

P. Corke, F. Spindler, François Chaumette

Combining Cartesian and polar coordinates in IBVS

Peter I. Corke* Fabien Spindler** Francois Chaumette***

Abstract— **Image-based visual servo (IBVS) is a simple, efficient and robust technique for vision-based control. Although technically a local method in practice it demonstrates almost global convergence. However IBVS performs very poorly for cases that involve large rotations about the optical axis. It is well known that re-parameterizing the problem by using polar, instead of Cartesian coordinates, of feature points overcomes this limitation. First, simulation and experimental results are presented to show the complementarity of these two parameterizations. We then describe a new hybrid visual servo strategy based on combining polar and Cartesian image Jacobians.**

I. Introduction

Image-based visual servoing (IBVS) is a robust and efficient means for controlling the pose of a robot based on visual features [1], [2]. While the technique is technically only a local method it has, in practice, demonstrated a very large field of convergence. For the last decade the visual servoing research community has investigated, and attempted to remedy, the few failure modes. The most notable of these modes was first reported in [3] and involves a set of point features and a pure rotation motion of π about that axis. In this case the camera, surprisingly, performs a pure translation along the optical axis to minus infinity. In fact for cases of large rotation, but less than π, about the optical axis the camera is observed to move away from the target as it rotates and then move back again — a phenomenon dubbed *camera retreat* [4]. This behaviour is non-obvious and it is also very inefficient and likely to cause robot joint limits to be exceeded. In [4] the camera retreat effect is explained intuitively by the fact that the IBVS control law causes feature points to move in straight lines on the image plane. For the case of pure rotation however, the points would *naturally* move along circular arcs. The consequence of this is that the camera scale must be changed dynamically so that the rotational motion appears as straight line motion — the scale change is achieved by the z-axis translation.

As already proposed in [5], this insight leads us to consider the use of polar coordinates where the required feature motion for pure camera rotation would be a straight line parallel to the θ axis. Similarly a pure scale change (z-axis translation) would cause radial motion of the points which corresponds to motion parallel to the r axis.

The contributions of this paper are an experimental analysis of the behavior of the system using the polar image Jacobian, a quantification of the image plane limits to control, and a new hybrid visual servoing scheme combining both Cartesian and polar image coordinates.

CSIRO ICT Centre, Brisbane, Australia. peter.corke@csiro.au
**EPI Lagadic, Inria-Rennes fabien.spindler@irisa.fr
***EPI Lagadic, Inria-Rennes francois.chaumette@irisa.fr

The next section, Section II, recalls the Image Jacobian for the classical IBVS and the polar form of IBVS which we call IBVS-P. Section III presents simulation and experimental results for various translational and rotational motion cases and also a general motion case. Section IV describes constraints on image plane motion which are more restrictive than for IBVS. Section V shows the performance of a hybrid Jacobian, built from the classic Cartesian optical-flow equation and the polar optical-flow equations, for the pure rotation case and the general motion case. Finally, in Section VI, we summarize and touch on current work that extends IBVS-P.

II. Image Jacobians for IBVS and IBVS-P

Assume that the camera is moving with translational velocity $\mathbf{T} = [T_x, T_y, T_z]$ and angular velocity $\Omega = [\omega_x, \omega_y, \omega_z]$ in the camera frame. Consider a world point, $\mathbf{P}$, with camera relative coordinates $\mathbf{P}c = [x, y, z]^T$.

Assuming a standard projective camera with unit focal length, our point feature $f = (u, v)$ is the image plane coordinates of the image feature where

$$u = \frac{x}{z},\ v = \frac{y}{z}. \tag{1}$$

We recall that the classical Jacobian $\mathbf{J_c}$, defined such that

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \mathbf{J_c}\mathbf{v} \tag{2}$$

where $\mathbf{v} = [T_x, T_y, T_z, \omega_x, \omega_y, \omega_z]$ is the instantaneous camera velocity, is given by [1]

$$\mathbf{J_c} = \begin{bmatrix} -1/z & 0 & u/z & uv & -(1+u^2) & v \\ 0 & -1/z & v/z & 1+v^2 & -uv & -u \end{bmatrix} \tag{3}$$

In polar coordinates our point feature $f = (r, \theta)$, comprising the radius of the feature point with respect to the optical centre

$$\begin{aligned} r &= \sqrt{u^2+v^2} \\ &= \frac{1}{z}\sqrt{x^2+y^2} \end{aligned} \tag{4}$$

and the angle

$$\begin{aligned} \theta &= \tan^{-1}\frac{v}{u} \\ &= \tan^{-1}\frac{y}{x}. \end{aligned} \tag{5}$$

The two feature representations are related by

$$u = r\cos\theta,\ v = r\sin\theta. \tag{6}$$

After simple calculations, it is possible to obtain the analytical form of the Jacobian $\mathbf{J_p}$ of feature $f = (r, \theta)$. It is given by [6], [7]

$$\mathbf{J_p} = \begin{bmatrix} \frac{c}{z} & \frac{s}{z} & -\frac{r}{z} & -(1+r^2)s & (1+r^2)c & 0 \\ -\frac{s}{rz} & \frac{c}{rz} & 0 & -\frac{c}{r} & -\frac{s}{r} & 1 \end{bmatrix} \quad (7)$$

where $c = \cos\theta$ and $s = \sin\theta$, which defines the polar optical flow equation

$$\begin{bmatrix} \dot{r} \\ \dot{\theta} \end{bmatrix} = \mathbf{J_p}\mathbf{v}. \quad (8)$$

Note that the features exactly considered in [6] were not the polar coordinates (r, θ), but $(r, r\theta)$. On one hand, this allows dealing with the problematic case where an image point is near the image centre where $r = 0$, in which case θ is not defined and four elements of the second line of the Jacobian (7) are infinite. On the other hand, it does not allow the simple and nice form (7) to be obtained. Indeed the Jacobian (7), is notable in that it has three constant elements. In the first row the zero indicates that radius is invariant to rotation ω_z around the optical axis. In the second row the zero indicates that polar angle is invariant to translation T_z along the optical axis (points move along radial lines), and the one indicates that the angle is directly proportional to camera rotation around the optical axis. As for the Cartesian point features (see (3)), the translational part of the Jacobian (the first 3 columns) are proportional to $1/z$. By comparing the two Jacobians (3) and (7), it is clear that they are complementary: u-image coordinates are invariant to translation T_y while v-image coordinates are invariant to translation T_x. This is the main idea at the basis of the hybrid control scheme that will be presented in Section V.

For control purposes we follow the normal procedure of computing one 2×6 Jacobian for each of N feature points and stacking them to form a $2N \times 6$ matrix

$$\begin{bmatrix} \dot{r}_1 \\ \dot{\theta}_1 \\ \vdots \\ \dot{r}_N \\ \dot{\theta}_N \end{bmatrix} = \mathbf{J}\mathbf{v} \quad (9)$$

where

$$\mathbf{J} = \begin{bmatrix} \mathbf{J_{p_1}} \\ \vdots \\ \mathbf{J_{p_N}} \end{bmatrix} \quad (10)$$

The control law is

$$\mathbf{v} = \mathbf{J}^+ \dot{f}^* \quad (11)$$

where $\dot{f}^*$ is the desired velocity of the features. Typically we choose this proportional to feature error

$$\dot{f}^* = -\gamma(f - f^*) \quad (12)$$

where γ is a positive gain, f is the current value of the feature vector, and f^* is the desired value, which leads to linear motion of features within the feature space. However for the polar coordinate case, since $\theta \in S$ rather than $\Re$ we define the error as

$$\dot{f}^* = f \ominus f^* \quad (13)$$

where $\ominus$ is modulo 2π subtraction for the angular component.

For Cartesian point features the feature coordinates are both distances on the image plane. In polar coordinates one feature coordinate is a distance and the other an angle.

Target points	$(\pm 0.25, \pm 0.25, 3)$
Focal length λ	5
Image plane bounds	$\pm 0.3, \pm 0.3$
Gain γ	0.001

TABLE I
SIMULATION PARAMETERS.

III. RESULTS

A. Simulation

We present simulation results for the cases:

1) Pure x-axis translation
2) Pure z-axis rotation
3) General motion about all axes.

The simulation parameters are summarized in Table I. Note that we treat image plane coordinates as distances rather than scaling them to pixels. Further, we assume that the depth of each feature point is known exactly. For each simulation we present the feature paths in the $u - v$ and the $r - \theta$ feature space, where the initial coordinate is marked with a 'o' and the final coordinate marked with a '*'. We also present the time history of the feature error, $\dot{f}^*$, and the demanded camera velocity screw components $\mathbf{v}$.

Figure 1 shows the case of pure x-axis translation. This results in slightly curved motion on the image plane and straight line motion on the polar image plane. The velocity curve shows that motion has occured along 3 DOF: x-translation, y-rotation and z-translation. This indicates incorrect decoupling in the linearization of the plant which does not occur for IBVS.

Figure 2 shows the case of pure z-axis rotation which is the classically difficult case exhibited in [3]. This results in circular motion on the image plane and horizontal (constant r) motion on the polar image plane with the only motion being about the z-axis. Note that some of the angles have wrapped around.

Finally, Figure 3 shows the case of rotation and translation about all axes. This results in non-straight motion on both the Cartesian and polar image planes.

B. Experiment

Experiments were conducted on a 6 d.o.f. gantry robot. A firewire Dragonfly2 camera with a resolution of 640×480 is mounted on the robot end effector. Images are acquired at 60Hz. The eye-to-hand transformation is calibrated and the camera intrinsic parameters are also calibrated: principal point coordinates (u0 , v0) = (310.2, 260.8), pixel ratio px

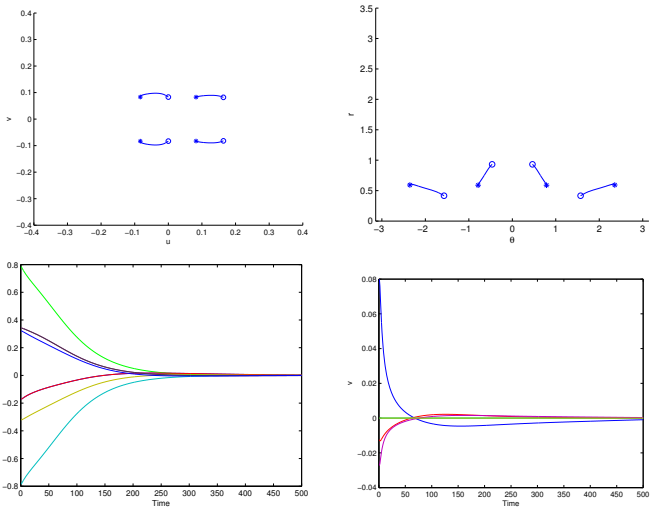

Fig. 1. Simulated pure x-axis translation of 0.2 with IBVS-P.

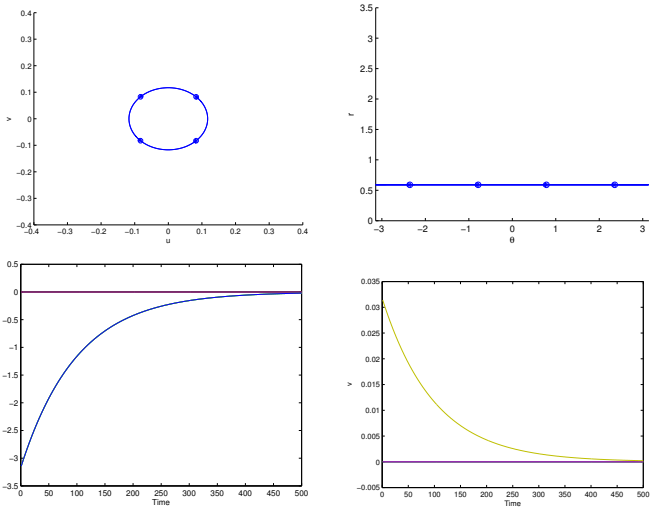

Fig. 2. Simulated pure z-axis rotation of π with IBVS-P.

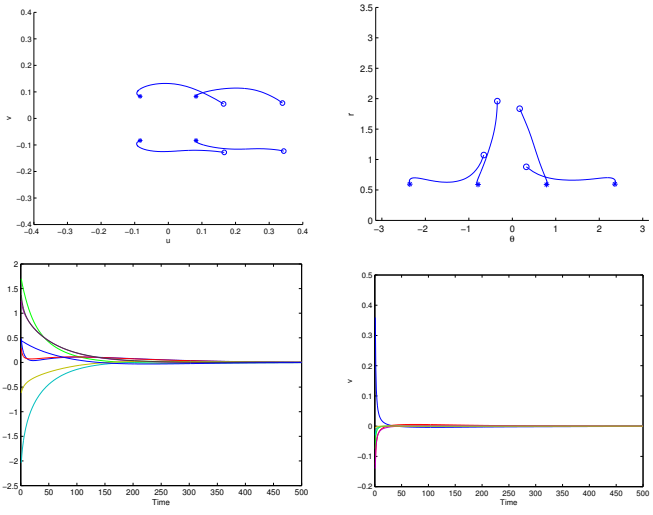

Fig. 3. Simulated general motion with IBVS-P.

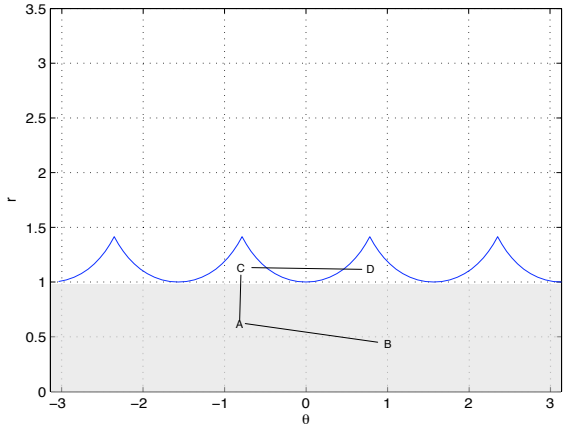

Fig. 4. Image plane in $r-\theta$ space.

= py = 1090 and distorsion kd = 0.22. In the experiments, the image Jacobian is computed based on the actual point depth based on pose estimation. The target is a 10×10cm square. Image processing consists of tracking 4 dots. From the center of gravity of the dot we compute the cartesian or polar coordinates of the COG in the image plane which are the features used for servoing. The desired position of the features is set so that the final camera pose is (0, 0, 0.6, 0, 0, 0). The target is then centered in the camera plane.

To be able to compare the behavior of the proposed scheme with IBVS, we first show Figure 6 results obtained for a general motion using IBVS. Figure 7 shows results for the same general motion using IBVS-P and we can see the desired convergence property. Results are not significantly different. Whereas we observe a better decrease of the velocities with IBVS-P. With IBVS we observe the bad behavior of the ω_y angular velocity which initially increases before decreasing. This indicates that the 3D trajectory of the camera is better with IBVS-P than with IBVS.

IV. Image Plane Limits to Control

Figure 4 shows the perimeter of the square image plane in polar coordinates. The scalloped sections are $1/\cos\theta$ shaped. As already mentioned the IBVS control law drives features along straight lines in the $r-\theta$ space. This presents no problems at all for feature trajectories within the shaded region indicated in Figure 4, for instance from point A to point B. However problems arise in the case of motion from point C to point D, since the feature will leave the image plane. Not that motion from A to C is quite feasible.

The shaded region in Figure 4 corresponds to the largest enscribed circle within the Cartesian image plane, and within which arbitrary feature motion can occur. However motion from one corner of the Cartesian plane to another is not feasible. However the potential field technique introduced in [4] could easily be integrated into this scheme to override the z-axis translation when feature points approach the edge of the image plane. For wide angle images from fisheye or catadioptric cameras the image is often circular in which case

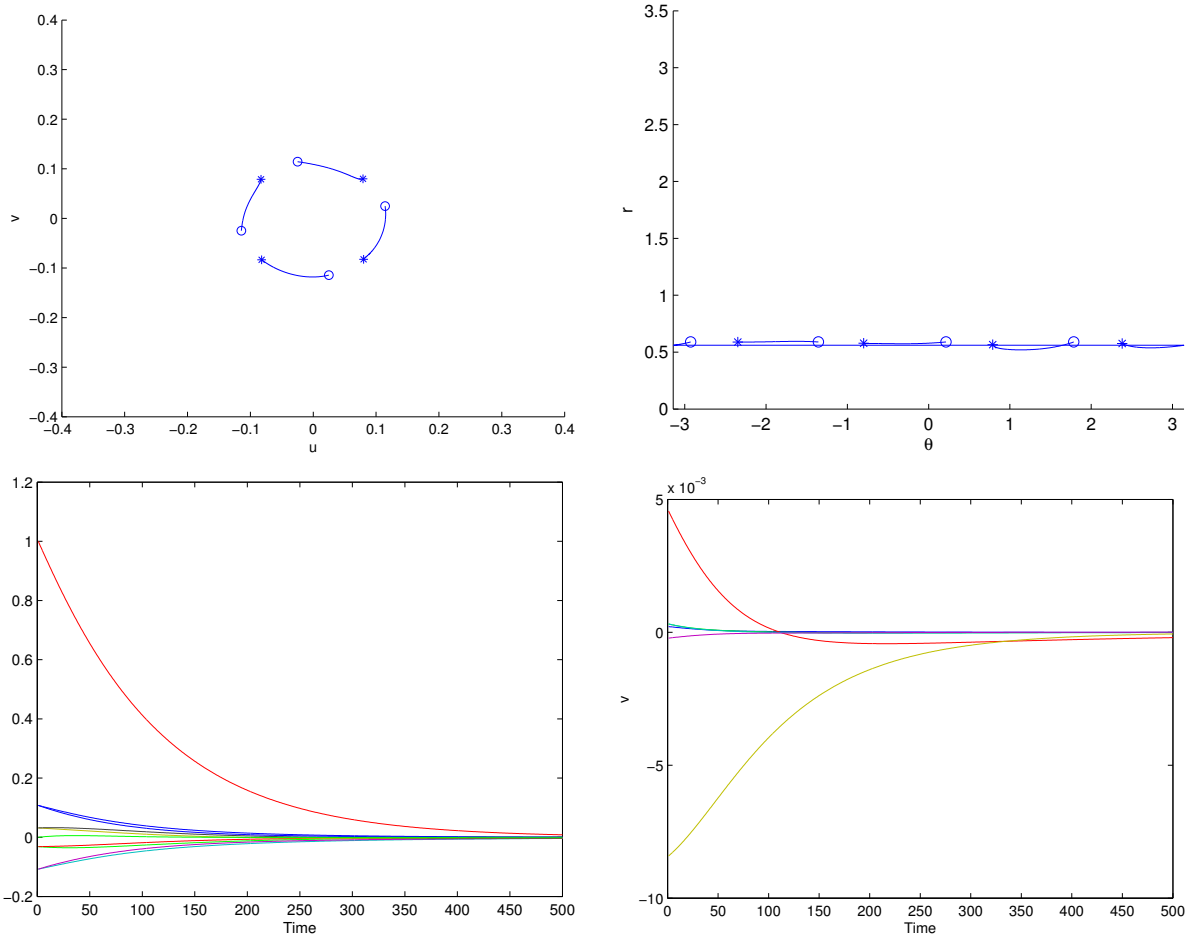

Fig. 5. Simulated hybrid Jacobian for pure z-axis rotation of π.

this constraint will not be problematic.

V. Hybrid image Jacobian

IBVS-P has almost complementary performance characteristics to IBVS. IBVS-P gives excellent performance for the case in which IBVS fails totally and acceptable performance for other cases although with poorer decoupling of x- and y-axis rotational and translational motions. This naturally suggests that some form of hybrid strategy could be used.

Previously proposed hybrid visual servo schemes [7] have partitioned the degrees of freedom. Here we propose to exploit the complementarity of IBVS and IBVS-P by using the image Jacobian to combine them. [6] also combined these two strategies by taking a linear combination of the two control laws where the weights were adjusted as a function of camera retreat [4].

In Section II we followed the standard practice in visual servoing by stacking polar optical flow blocks for each feature point, as defined by (7), in order to create the image Jacobian. For classical IBVS we would stack Cartesian optical flow blocks. Instead, here we propose a scheme that stacks both kinds of blocks to create a hybrid image Jacobian. There are two possibilities for that. The first one consists in choosing a representation for each point (Cartesian or polar), while the second one consists in combining both representations (Cartesian and polar), leading to a highly redundant system. Indeed, in that case, for N feature points we could stack upto N polar and N Cartesian Jacobian blocks, leading to a global Jacobian with $4N$ lines. Both appraoches are tested in the following.

Figure 5 show a simulation in which the hybrid Jacobian comprises the Cartesian Jacobian blocks for points 1 and 2 and the polar Jacobian blocks for points 2 and 3. We can see that the motion of the points is somewhere between a circle and a straight line. The dominant velocity is z-axis rotation and with a small amount of camera retreat. For x- and y-axis rotation the hybrid scheme produces acceptable feature motion but with even more cross-coupling than for IBVS-P.

Figure 8 show experimental results for the general motion case where the hybrid Jacobian comprises the Cartesian and polar Jacobian blocks for each of the points, that is, eight Jacobian blocks resulting in a 16×6 image Jacobian. As for IBVS-P results presented Figure 7, and unlike results Figure 6 obtained with IBVS, the camera velocities have a good behavior. The 3D trajectory of the camera is also better with hybrid Jacobian than with IBVS. Results using the Cartesian Jacobian blocks for points 1 and 2 and the polar Jacobian blocks for points 2 and 3 led to robot joint angle limits, and

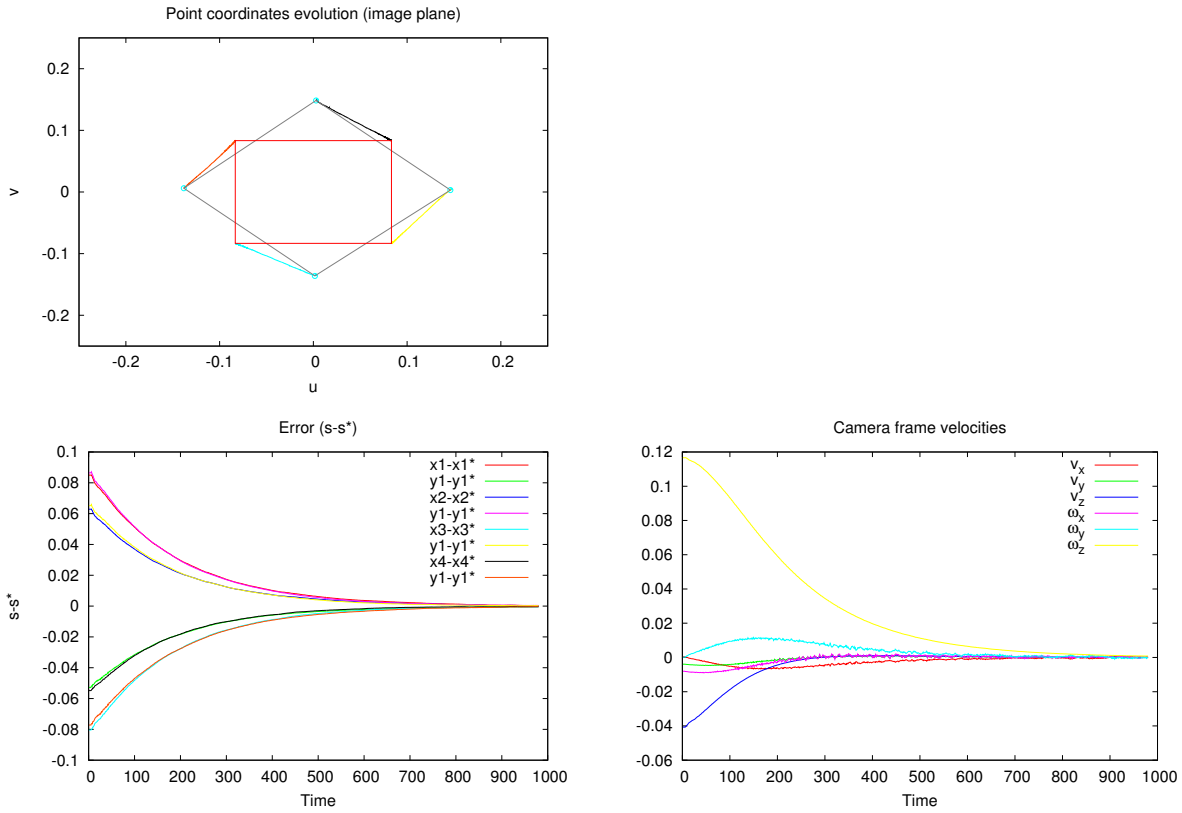

Fig. 6. Experimental results with IBVS for general motion.

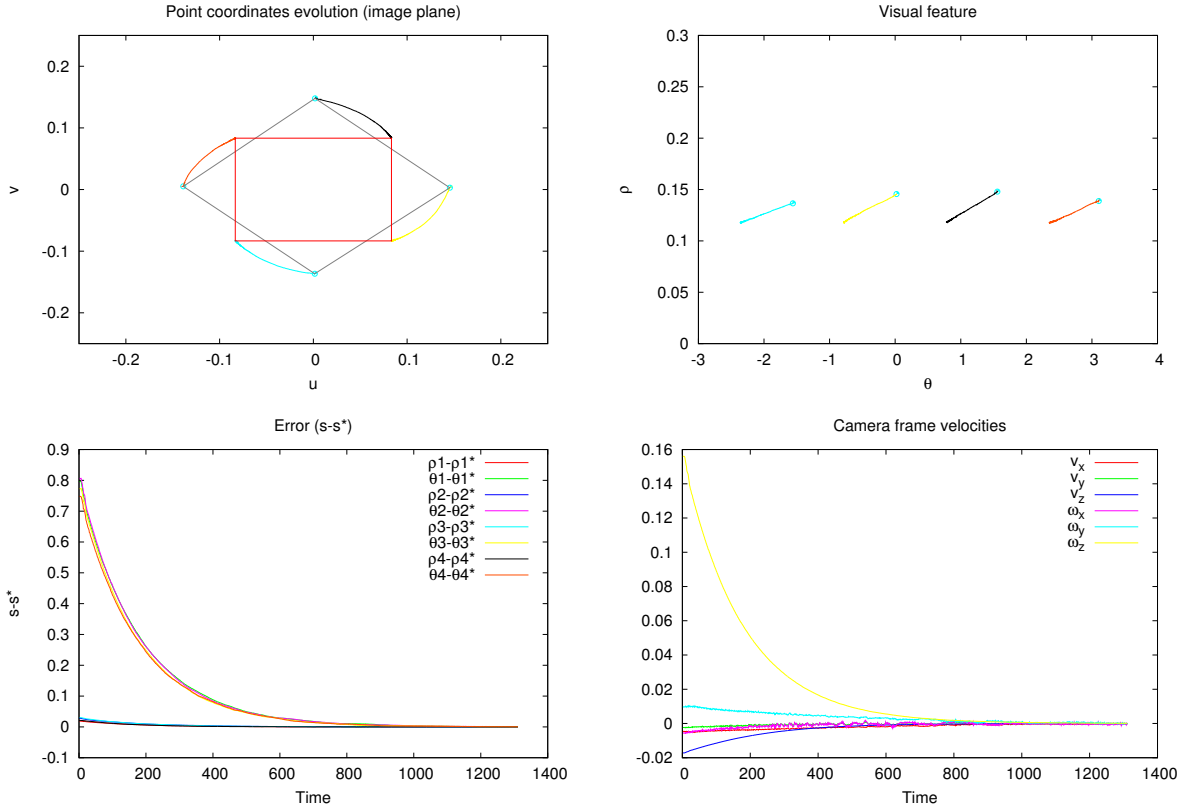

Fig. 7. Experimental results with IBVS-P for general motion.

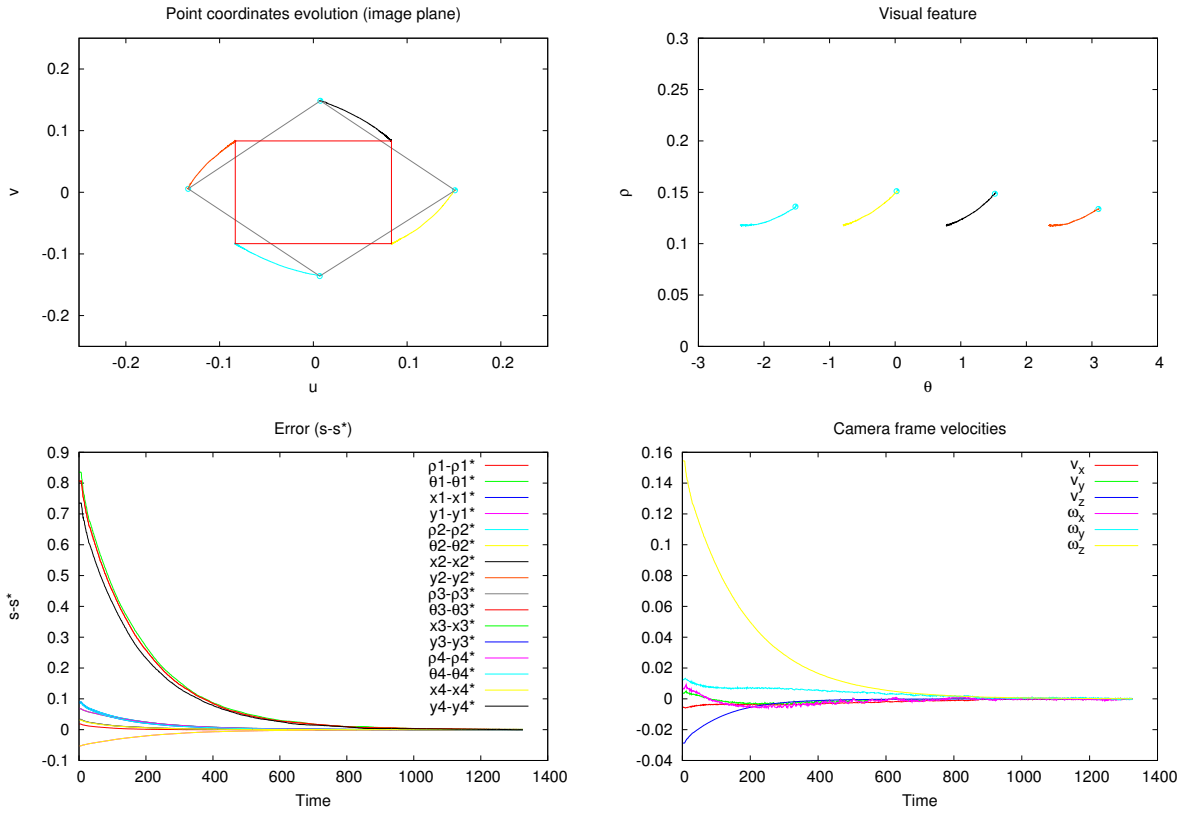

Fig. 8. Experimental results for hybrid Jacobian for general motion.

this phenomena is the subject of further work.

VI. Conclusion

We have seen how re-parameterizing the IBVS problem by using polar, instead of Cartesian coordinates, can improve performance for large optical axis rotation. Performance for other cases is acceptable though there is poorer decoupling of x- and y-axis rotation and translational motions, and this requires further work to understand its cause.

A new hybrid IBVS/IBVS-P visual servo strategy was introduced based on stacking polar and Cartesian image Jacobian blocks and was shown to give better performance for large z-axis rotations than pure IBVS. Current work is investigating the effect of different feature point depth estimation strategies on IBVS-P performance, and the effect of different ways of combining polar and Cartesian Jacobian blocks.

References

[1] S. Hutchinson, G. Hager, and P. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, pp. 651–670, Oct. 1996.

[2] F. Chaumette and S. Hutchinson, "Visual servo control. i. basic approaches," *Robotics & Automation Magazine, IEEE*, vol. 13, pp. 82–90, Dec. 2006.

[3] F. Chaumette, "Potential problems of stability and convergence in image-based and position-based visual servoing," in *The confluence of vision and control* (D. Kriegman, G. Hager, and S. Morse, eds.), vol. 237 of *Lecture Notes in Control and Information Sciences*, pp. 66–78, Springer-Verlag, 1998.

[4] P. Corke and S. A. Hutchinson, "A new partitioned approach to image-based visual servo control," *IEEE Trans. Robot. Autom.*, vol. 17, pp. 507–515, Aug. 2001.

[5] M. Iwatsuki and N. Okiyama, "Rotation-oriented visual servoing based on cylindrical coordinates," in *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, vol. 4, pp. 4198–4203 vol.4, 2002.

[6] M. Iwatsuki and N. Okiyama, "A new formulation of visual servoing based on cylindrical coordinate system with shiftable origin," in *IEEE/RSJ Int. Conf. Intelligent Robots and System.*, vol. 1, pp. 354–359, 2002.

[7] F. Chaumette and S. Hutchinson, "Visual servo control. ii. advanced approaches [tutorial]," *Robotics & Automation Magazine, IEEE*, vol. 14, pp. 109–118, March 2007.