A Shape Tracking Algorithm for Visual Servoing

Peihua Li*, François Chaumette and Omar Tahri
IRISA/INRIA Rennes
Campus de Beaulieu, 35 042 Rennes-cedex, France
Firstname.Name@irisa.fr

***Abstract*— The paper contributes to presenting both an accurate and robust shape tracking algorithm and a novel visual servoing method. Two steps are involved in the tracking algorithm. Firstly the object shape is assumed to vary under an af£ne model, and the edge detection is performed along the normal lines to the contour. As a result it is possible to use a Kalman £lter to perform ef£cient tracking. The second step concerns image matching based on perspective model, which is achieved iteratively by searching locally along the normal lines also. As to visual servoing we propose to control the translations of the robot with the normalized zeroth and £rst order image moments, and to control the orientation with rotation axis and angle extracted from a Homography matrix. Two experiments demonstrate that the tracking algorithm is accurate and robust enough to be used in visual servoing, and the novel visual servoing method is superior to traditional ones.**

***Index Terms*— Object tracking, Kalman £lter, Visual servoing.**

I. Introduction

Visual servoing is usually concerned with the issue of controlling the six degree of freedom (d.o.f) of the robot to a desired position, provided the visual information as a feedback achieved from a camera usually installed on the end effector of the robot [1]. Embedded in the control system, the visual information is required to be accurate, robust and in real-time. How to get this knowledge from image sequences captured by a camera is a dif£cult and challenging problem, in which visual tracking is generally regarded as being playing an important role [2], [3]. These works are, however, either interested in simple image features such as corners, lines or regions, or have an assumption that the 3D structure of the object is known. In contrast, we consider in this paper realistic planar objects having natural shapes. To track objects robustly in complex background, non-linear £ltering algorithms, say, particle £lter [4] or unscented particle £lter [5] are good choices. Unfortunately they are not £t for visual servoing task, due to their computational ef£ciency and large tracking errors.

When we have got knowledge of the interested object, either by tracking technology or binary image, another issue is how to develop *visual features*, the derivative of which, the so-called interaction matrix, establishes links between image and velocity screw of the robot. The closed-form interaction matrix related to structured visual features, such as points, straight lines, cylinders, etc. are derived in [6]. Recently image moments related to objects with natural shapes are proposed to control six d.o.f robot [8], [9], which has advantage of decoupling the system. Malis et al. [10] proposes 2 1/2D method to overcome the disadvantages of 2D and 3D visual servoing while combining their respective advantages. Unfortunately, all of these works have the assumption that the images captured are binary and object shapes can be achieved accurately. These are generally not cases in real-world. When we use tracking technology to achieve object shapes, the tracking errors usually exist, especially when the background is complex. It is well known that the higher image moments are sensitive to noise and errors, so moments based visual servoing will probably not work when the background becomes challenging. For 2 1/2D method, four of six visual features are related to 3D of the object, although the whole object generally will not go out of the view of the camera, part of it may do. Then occlusions may occur, which is a threat to tracking algorithms and thus not desirable in visual servoing.

Our paper has contributions on both aspects discussed above. For the tracking aspect, we develop Kalman £lter and image matching to track object accurately and robustly. The Kalman £lter based tracking is developed on the basis of the work introduced in [7]. The shape of the object is described with B-Spline curves. We £rst assume that the shape of the object varies during visual servoing process according to an af£ne model, which leads to a six dimensional linear system whatever the number of control points may be. However, the af£ne model is not accurate and to overcome this, we introduce image matching based on perspective model on the basis of tracking result. This issue concerns matching of the desired curve and the current curve and is a nonlinear problem. For ef£ciency we simplify it as a linear one, according to local search of the edge points along normal lines to the contour. The image matching will in general converge during within less than ten iterations in less than 10 milliseconds.

For the development of visual features, we propose a new strategy, using the normalized zeroth and the £rst image moments to control the translation of the camera, while using rotation vector extracted from Homography matrix to control the orientation of the camera. With this strategy,

*Current address: College of Computer Science and Technology, Heilongjiang University, Heilongjiang Province, 150001, China, peihualj@hotmail.com

we have seen a signi£cant improvement of visual servoing performance over classical 2 1/2D method. On the one hand, the lower moments are not sensitive to errors; on the other hand, using three 2D visual features to control translation we avoid the occlusion which may occur in classical 2 1/2D while maintaining its advantage.

A similar work to ours is described in [11], which is also concerned with visual servoing with respect to planar objects. The differences are, however, signi£cant in tracking algorithm, image matching method and visual servoing.

The remainder of the paper is organized as follows. In Section 2, we describe shape tracking based on af£ne model, involving shape representation, motion model and measurement model, and image matching based on perspective model and local search along normal lines. Section 3 introduces the improved 2 1/2D visual servoing method, using the normalized low order image moments and rotation matrix extracted from a homography matrix. Two experiments are given to demonstrate the effectiveness of the tracking algorithm and the new visual servoing method in Section 4, comparisons between visual servoing based on image moments and classic 2 1/2D method being also made. The concluding remarks are given at last.

II. Tracking Algorithm Based on Kalman Filter and Image Matching

A. Shape representation with B-Spline curves

The desired shape of the object (the desired curve) is parameterized as a B-spline curve as follows

$$\mathbf{r}_*(s) = \mathbf{H}_*(s)\mathbf{Q}_* \tag{1}$$

that is:

$$\begin{bmatrix} x_*(s) \\ y_*(s) \end{bmatrix} = \begin{bmatrix} \mathbf{B}_*(s) & 0 \\ 0 & \mathbf{B}_*(s) \end{bmatrix} \begin{bmatrix} \mathbf{Q}_*^x \\ \mathbf{Q}_*^y \end{bmatrix}$$

where $\mathbf{B}_*(s) = [b_0(s) \quad \cdots \quad b_{N_B-1}(s)]$, for $0 \le s \le L$, $b_i(s)$ is the ith B-spline basis function, $\mathbf{Q}_*^x$ is a column vector consisting of x coordinates of all the control points and so is $\mathbf{Q}_*^y$, and L is the number of spans. We assume that, at the £rst step of tracking, the shape of the object at the current time step (the current curve) varies under af£ne model as below

$$\mathbf{r}(s) = \mathbf{A}\mathbf{r}_*(s) + \mathbf{t} \tag{2}$$

that is:

$$\begin{bmatrix} x(s) \\ y(s) \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{bmatrix} x_*(s) \\ y_*(s) \end{bmatrix} + \begin{bmatrix} a_5 \\ a_6 \end{bmatrix}$$

Combining Eqs. (1) and (2), and note that $\sum_{i=0}^{N_B-1} b_i(s) = 1$ for all s, we get the following equation

$$\mathbf{Q} = \mathbf{W}\mathbf{X} + \mathbf{Q}_* \tag{3}$$

where $\mathbf{Q}$ is a vector containing the x and y coordinates of the control points of the current curve, $\mathbf{W} = \begin{bmatrix} \mathbf{1} & \mathbf{0} & \mathbf{Q}_*^x & \mathbf{0} & \mathbf{0} & \mathbf{Q}_*^y \\ \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{Q}_*^y & \mathbf{Q}_*^x & \mathbf{0} \end{bmatrix}$ and $\mathbf{X} = \begin{bmatrix} a_5 & a_6 & a_1 - 1 & a_4 - 1 & a_3 & a_2 \end{bmatrix}$.

B. Motion Model, Measurement Model and Kalman Filter

The motion equation of the system is modelled as the multi-dimensional second order auto-regression (AR) process, which generally can be seen as the discretized form of a continuous stochastic second order dynamic system. This multi-dimensional AR process may be regarded as the direct extension of a 1D AR process, which has the following form

$$x_k = a_1 x_{k-1} + a_0 x_{k-2} + b_0 \nu \tag{4}$$

where $a_1 = -\exp(-2\beta\tau)$, $a_0 = 2\exp(-\beta\tau)\cos(\omega\tau)$, $b_0 = \sqrt{1 - a_1^2 - a_0^2 - 2\frac{a_1 a_0^2}{1-a_1}}$, ν is one dimensional Gaussian i.i.d. noise, β, ω and τ are the damping coef£cient, the oscillation period and the sampling period of the corresponding continuous system. It is desirable, in practice, to model the translation and the shape variations of the contour separately, so the 1D AR process is extended respectively to two complementary subspaces of the shape space: translation subspace and deformation subspace. Then the multi-dimensional motion model can be represented as below

$$\mathbf{X}_k = \mathbf{A}_1\mathbf{X}_{k-1} + \mathbf{A}_0\mathbf{X}_{k-2} + \mathbf{B}_0\mathcal{V} \tag{5}$$

where $\mathcal{V}$ is multi-dimensional Gaussian i.i.d.

The normals of a £nite number of sample points on the contour are searched for features. The length of these normals, termed measurement lines, are determined by the covariance of the Kalman £lter and thus vary adaptively. A 1D Canny edge detector is applied to each measurement line and the points of local maximum are adopted as detected features. The measurement procedure involves computing measurement lines, discretising these lines, extracting grey-level values from images, and make convolutions with 1D Canny kernel. The measurement equation can be described as below

$$\upsilon_k = \mathbf{H}(\hat{\mathbf{X}}_k - \mathbf{X}_k) + \omega_k \tag{6}$$

where $\hat{\mathbf{X}}_k$ is the prediction using Equ. (5), υ_k is the innovation, $\mathbf{H} = \begin{bmatrix} \hat{n}(s)^\top \bigotimes \mathbf{B}_*(s) & \mathbf{O} \end{bmatrix} \mathbf{W}$, $\hat{n}(s)$ is the normal to the contour represented by $\hat{\mathbf{X}}_k$ and ω_k is Gaussian whose variance is σ^2.

Now that we have motion model (5) and measurement model (6), a set of Kalman prediction equation and update equation can be computed iteratively to estimate the current system state when each new frame is available.

C. Image Matching

The tracking algorithm based on af£ne model and edge detection along normal lines to the contour are very stable when the perspective effects are not strong. Yet the tracking

behavior becomes instable when the af£ne model assumption is violated. In addition, even if the tracking succeeds, the visual servoing based on a tracking using an af£ne model will not be able to control the 6 robot d.o.fs, since the af£ne model allows to control correctly only 4 robot d.o.fs. We turn to image matching under full perspective model to achieve more accurate tracking results. The image matching between the desired curve $\mathbf{r}_*(s)$ and the current curve $\mathbf{r}_k(s)$ is an optimization problem as follows

$$\arg\min_{\mathbf{A},\mathbf{t}} \|\mathbf{r}_*(s) - \mathbf{r}'_k(s)\| \tag{7}$$

where $\| \bullet \|$ denotes L_2 norm, and $\mathbf{r}'_k(s)$ satis£es the following equation,

$$\mathbf{r}'_k(s) = \frac{\mathbf{A}\mathbf{r}_k(s) + \mathbf{t}}{a_7 x_k(s) + a_8 y_k(s) + 1}$$

For the computational ef£ciency, we do not solve the non-linear optimization problem described by Eq. (7). Instead, we consider the following equation

$$x_k(s_i) = \frac{a_1 x_*(s_i) + a_2 y_*(s_i) + a_5}{a_7 x_*(s_i) + a_8 y_*(s_i) + 1} \tag{8}$$

$$y_k(s_i) = \frac{a_3 x_*(s_i) + a_4 y_*(s_i) + a_6}{a_7 x_*(s_i) + a_8 y_*(s_i) + 1} \tag{9}$$

where $(x_k(s_i), y_k(s_i))$ and $(x_*(s_i), y_*(s_i))$ are respectively the sampled points on the current curve and the desired curve. We can arrange the above equation, and get the following linear equation

$$\begin{bmatrix} x_*(s_i) & y_*(s_i) & 0 & 0 & 1 & 0 & -x_*(s_i)x_k(s_i) & -y_*(s_i)x_k(s_i) \\ 0 & 0 & x_*(s_i) & y_*(s_i) & 0 & 1 & -x_*(s_i)y_k(s_i) & -y_*(s_i)x_k(s_i) \end{bmatrix} \mathbf{a} = \begin{bmatrix} x_k(s_i) \\ y_k(s_i) \end{bmatrix} \quad i = 1, \cdots, N_s \tag{10}$$

where $\mathbf{a} = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 \end{bmatrix}^T$, and N_s is the number of the sampled points. The image matching algorithm is described as follows.

1 Sample the desired curve to obtain $(x_*(s_i), y_*(s_i)), i = 1 \cdots N_s$. Given the initial value of $a_1, \cdots, a_6$ from the tracking result, and set $a_7 = a_8 = 0$.
2 Compute the corresponding points of the desired curve on the current curve according to Equ. (8) and (9), and £t these points to get a B-spline curve $\mathbf{r}'_k$.
3 Re-sample the curve $\mathbf{r}'_k$ to get new sample points $(x'_k(s_i), y'_k(s_i))$. Search along the normal lines as described in subsection II-B, obtaining the detected points $(x_k(s_i), y_k(s_i))$.
4 Calculate the vector $\mathbf{a}$ according to Eq. (10) with the Penrose-Moore generalized inverse of matrix.
5 Compute the norm $\|\mathbf{r}'(s) - \mathbf{r}_k(s)\|$. If it is less than a pre-de£ned threshold or maximum iteration number reaches, end the iteration; otherwise, go to step 2.

Fig. 1 shows two typical image matching results for a book and a leaf, in which Fig. 1.a and Fig. 1.c demonstrate the tracking result with af£ne model and Fig. 1.b and 1.d are corresponding matching results using the full perspective model. Matching errors versus iteration numbers are shown in Fig. 1.e and 1.f for the book and the leaf respectively. We can see that in both cases the matching algorithm converges within 10 iterations.

III. An Improved Visual Servoing Method

Our strategy consists in a novel combination of 2D and 3D knowledge of the object. More speci£cally, we use the zeroth and £rst moments (2D) to control the translation of the robot, and rotation matrix extracted from homography matrix (3D) to control the orientation of the robot. Our idea is very simple: if we can well control the translation of the robot, then the object will be kept within the £eld of view of the camera. Compared with the classical 2 1/2D visual servoing, the advantages are signi£cant. First, we avoid the situations that the object is out of the view, which often occurs with 2 1/2D method, although not signi£cantly like 3D visual servoing. Second, the interaction matrix with respect to moments have nice decoupling properties, in contrast with the interaction matrix with respect to the reference point adopted in [10], [11].

We consider that the planar closed object is parallel to the image plane for the desired con£guration. The zeroth and the £rst moments are the area a, and the coordinates x_c and y_c of the centroid of the object. Furthermore, de£ne

$$s_a = Z_* \sqrt{\frac{a_*}{a}},\ s_x = s_a x_c,\ s_y = s_a y_c \tag{11}$$

where a_* is the area of the desired curve, and Z_* is the desired depth between the camera and the object. The interaction matrix $\mathbf{L}_T$ related to these normalized features is given as below [9]

$$\begin{bmatrix} -1 & 0 & 0 & s_a\epsilon_{11} & -s_a(1+\epsilon_{12}) & s_y \\ 0 & -1 & 0 & s_a(1+\epsilon_{21}) & -s_a\epsilon_{11} & -s_y \\ 0 & 0 & -1 & -3s_y/2 & 3s_x/2 & 0 \end{bmatrix} \tag{12}$$

where $\epsilon_{11} = n_{11} - x_c y_c/2$, $\epsilon_{12} = n_{20} - x_c^2/2$, and $\epsilon_{21} = n_{02} - y_c^2$, and n_{11} and n_{02} are the normalized central moments of order two. We thus obtain the same dynamics for the three features and the three translational d.o.f, which leads to an adequate robot translational trajectory. Notice that the three translations are decoupled.

We control the orientation of the robot with the rotation vector extracted from homography matrix. Let $\mathbf{K}$ be the intrinsic matrix of the camera, then the pixel coordinates $\mathbf{p}$ is related to the normalized coordinates $\mathbf{m}$ with the following equation

$$\mathbf{p} = \mathbf{K}\mathbf{m} \tag{13}$$

In section II-C, we have achieved the vector $\mathbf{a}$ from which we get immediately the homography matrix $\mathbf{G}$ expressed

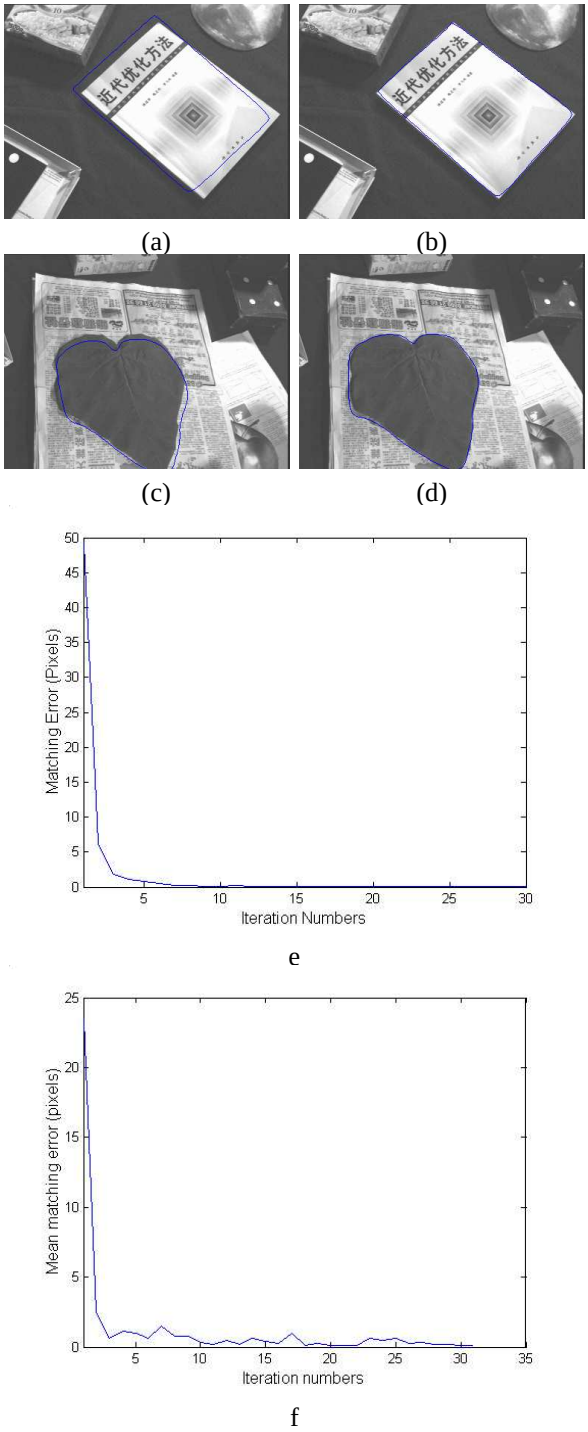

Fig. 1. Image matching for a book and a leaf. Tracking results for the two cases are shown in (a) and (c) respectively, and results after image matching in (b) and (d). Matching error versus iteration number are illustrated in (e) for the book and (f) for the leaf. We can see that in both cases the matching algorithm converges within 10 iterations.

in pixels. Assuming the camera calibration is known, the Euclidean homography matrix $\mathbf{H}$ is calculated as below

$$\mathbf{H} = \mathbf{K}^{-1}\mathbf{G}\mathbf{K} \tag{14}$$

The rotation matrix $\mathbf{R}$ between the desired view and the current view can be achieved from matrix $\mathbf{H}$ as introduced in [12]. The multiplication of rotation axis $\mathbf{u}$ and the rotation angle θ obtained from $\mathbf{R}$ is selected as visual features to control the rotation of the camera [10], the interaction matrix of which has the following form

$$\mathbf{L}_\omega = \left[\mathbf{0}_3 \quad \mathbf{I}_3 - \frac{\theta}{2}[\mathbf{u}]_\times + (1 - \frac{\mathrm{sinc}(\theta)}{\mathrm{sinc}^2\frac{\theta}{2}})[\mathbf{u}]_\times^2\right] \tag{15}$$

where $\mathrm{sinc}(\theta) = \sin(\theta)/\theta$, and $[\mathbf{u}]_\times$ is the antisymmetric matrix of the vector $\mathbf{u}$.

Now we have six visual features

$$\mathbf{s} = (s_x, s_y, s_a, \mathbf{u}\theta) \tag{16}$$

and its interaction matrix has the form $\mathbf{L_s} = (\mathbf{L_T}, \mathbf{L}_\omega)$. In this paper, the classical control law as below is used

$$\mathbf{v}_e = -\lambda \mathbf{L}_\mathbf{s}^{-1}\mathbf{e} \tag{17}$$

where $\mathbf{v}_e$ is the camera kinematic screw sent to the low level robot controller, λ is a positive gain, and $\mathbf{L}_s^{-1}$ is the inversive of the interaction matrix computed for the desired value, and $\mathbf{e} = \begin{bmatrix} s_x - s_{x*} & s_y - s_{y*} & s_a - s_{a*} & \mathbf{u}\theta \end{bmatrix}$ is the error of the visual features between the desired curve and the current curve, where s_{x*}, s_{y*} and s_{a*} are the coordinates of the centroid of and area of the desired curve.

IV. Experiments

To demonstrate the effectiveness of our framework, we make experiments with a six d.o.f eye-in-hand robot. The program is developed under Linux with C++. The tracking algorithm is initialized manually in the £rst frame.

The £rst experiment is concerned with visual servoing with respect to a book. The desired image and the initial image are given in Fig. 2.a and 2.b. Figs. 2.e and 2.h, Figs. 2.f and 2.i, and Figs. 2.g and 2.j show, during visual servoing process, the variations of the visual feature errors and the velocity of the robot, based on the image moments [9], 2 1/2D, and our proposed method respectively. Although the robot converges to the desired position with all of the three methods, it is very clear that the new method demonstrates better behavior than the other two, in the sense that in the former case both the feature errors and the speed of the robot decrease exponentially and is very smooth. We can also note that partial occlusions occur with 2 1/2D method during frame 20 to frame 71. Two typical frames (38 and 56) are shown in Fig 2.c and 2.d respectively, where red lines indicate the tracking result.

The second experiment is more challenging in two aspects. Firstly, the background is cluttered and shadows affect the accurateness of the edge detection while making measurement (both in tracking and in image matching), the

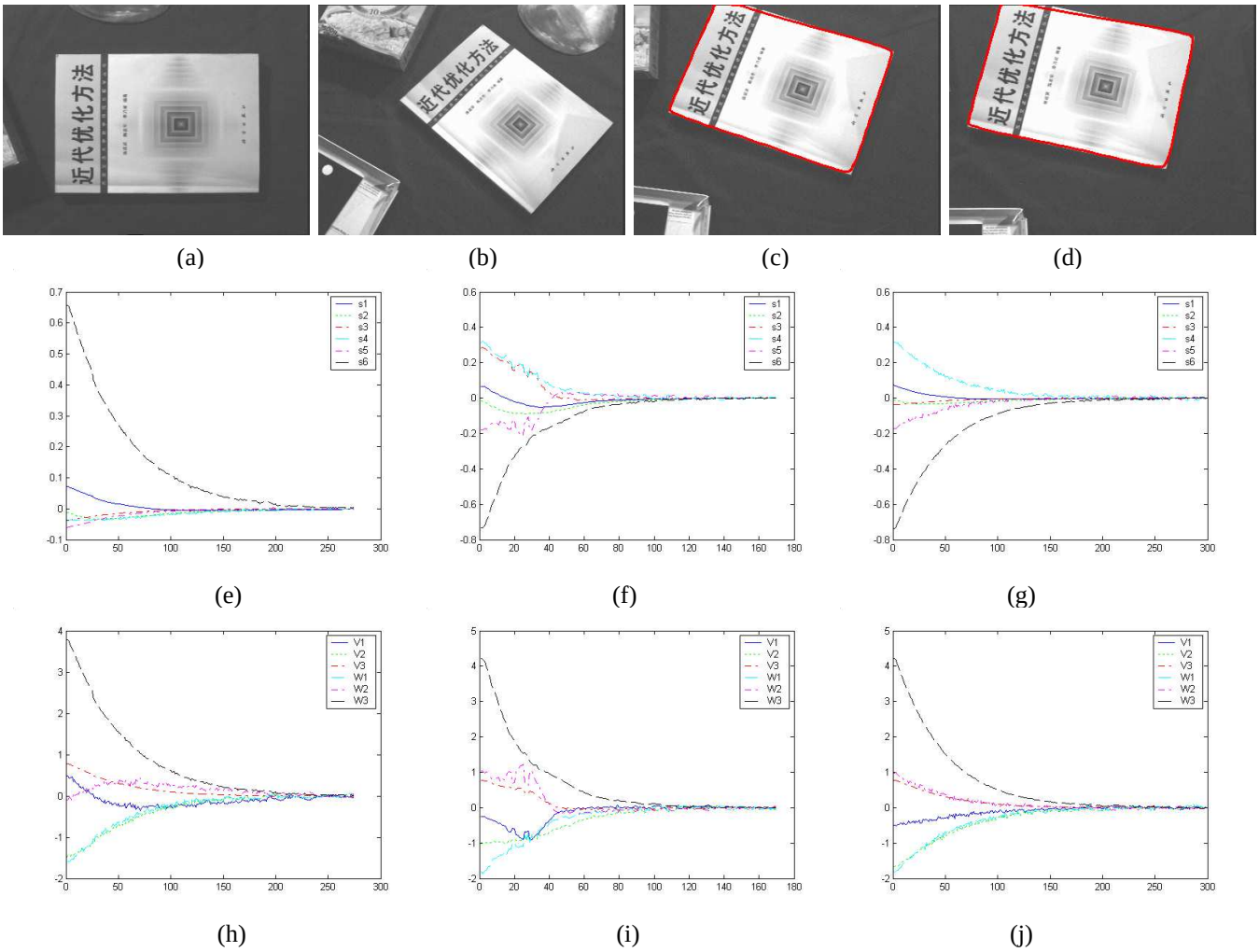

Fig. 2. Tracking and visual servoing with a book with the desired image (a) and the initial image (b). Shown in (c) and (d) are tracking results of frame 38 and 58 with 2 1/2D method, note the occlusion of the object. Feature errors (unit: m^2 for s1 and s2, m for s3, and rad for s4, s5, and s6) versus frame numbers are plotted based on image moments (e), 2 1/2D (g) and the new method (i). Camera velocity (cm/s and dg/s) versus frame number are illustrated based on image moments (f), 2 1/2D (h) and the new method (j).

tracking errors thus become larger than in the previous experiment. Also both the translation and rotation of camera are large between the desired and the initial position, as demonstrated in Fig. 3.a and 3.b. In this complex situation, visual servoing based on image moments fails to converge, as presented by Fig. 3.e and 3.h. It is due to the fact that higher order moments are not computed with suf£cient accuracy. We also note that, during visual servoing process, an occlusion occurs and is considerable from frame 68 to 306 when using 2 1/2D method. Two typical frames are shown in Fig. 3.c and 3.d. Despite the severe occlusion, it converges thanks to the robustness of our tracking algorithm to occlusion. In contrast, our proposed method behaves very well, reaching the desired position smoothly without any occlusion occurrence. From Fig. 3.g and 3.j, we can see clearly that the visual features and the speed of the robot based on the new method decrease exponentially and is very smooth, better than those based on 2 1/2D method.

V. Conclusions

In the paper, we £rstly present a novel shape tracking algorithm. Kalman £lter is used to locate object sequentially under af£ne model, followed by an image matching approach to get more accurate tracking results under full perspective model. The two stage tracking algorithm is necessary for robust and accurate tracking. One could think that under af£ne model, we can control the six d.o.f of the robot, which is wrong. The second stage overcomes the effect of strong perspective effects. On the basis of the novel tracking algorithm, we secondly propose an improvement strategy for visual servoing, in which the control of the translation and rotation are decoupled, and translation is controlled with the normalized zeroth and £rst order image moments and orientation with rotation matrix. Two experiments show that our tracking is accurate enough

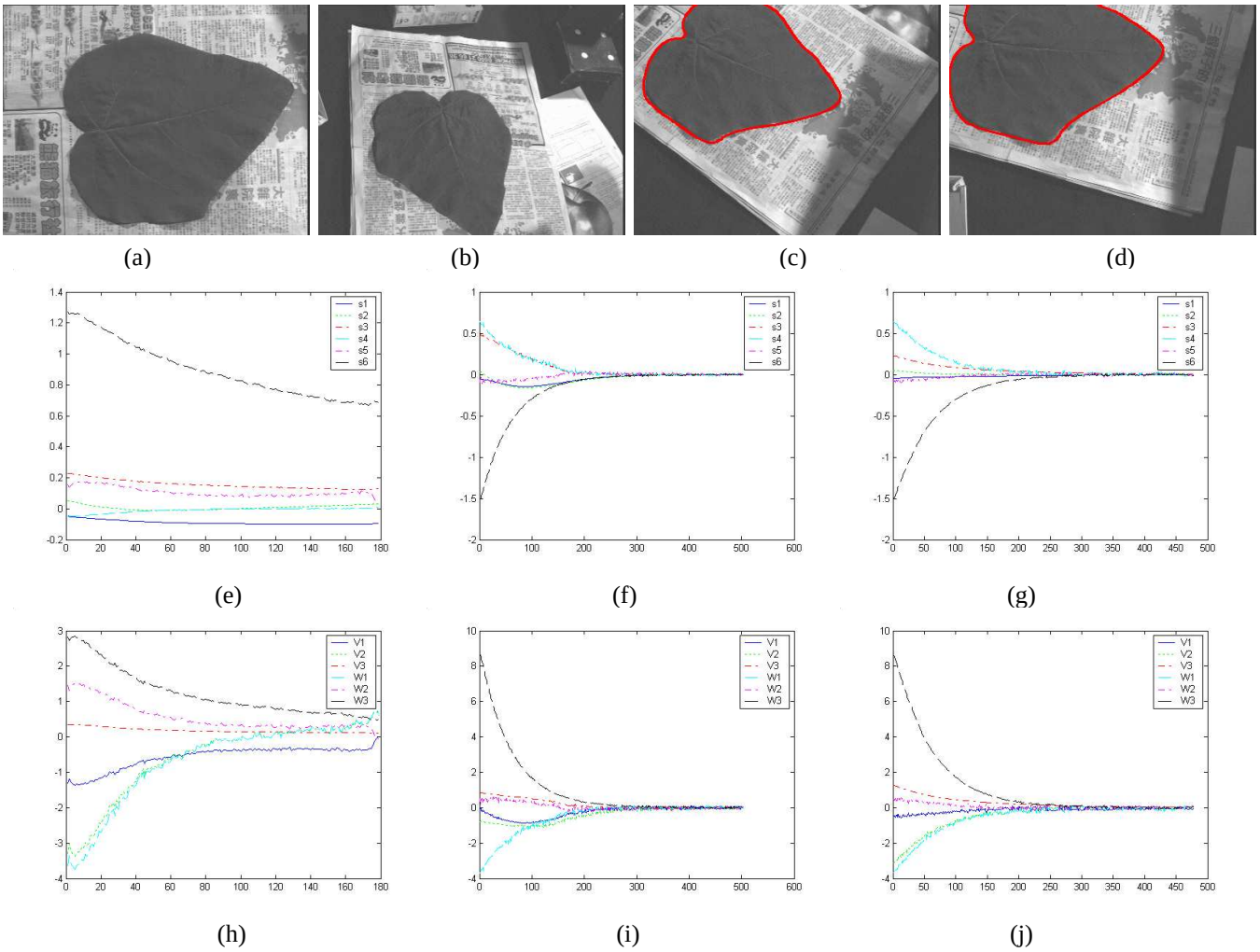

Fig. 3. Tracking and visual servoing with a leaf with the desired image (a) and the initial image (b). Feature errors (unit: m^2 for s1 and s2, m for s3, and rad for s4, s5, and s6) versus frame numbers are plotted based on image moments (e), 2 1/2D (g) and the new method (i). Camera velocity (cm/s and dg/s) versus frame number are illustrated based on image moments (f), 2 1/2D (h) and the new method (j). Shown in (c) and (d) are tracking results of frame 38 and 58 with 2 1/2D method, note the occlusion of the object is considerable.

for visual servoing in complex situations and robust to large occlusions. In addition, comparisons demonstrate that the new visual servoing method is better than the state of the art.

REFERENCES

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

[2] G.D. Hager, and K. Toyama, "The XVision system: a general-purpose substrate for portable real-time vision applications," Computer Vision and Image Understanding, vol. 69, no. 1, pp. 23–37, Jan. 1998.

[3] E. Marchand, "ViSP: a software environment for eye-in-hand visual servoing," In IEEE Int. Conf. on Robotics and Automation, ICRA'99, vol. 4, pp. 3224–3229, Detroit, Michigan, May 1999.

[4] P. Li, T. Zhang, Arthur E.C. Pece, "Visual contour tracking based on particle £lters," Image and Vision Computing, vol. 21, no. 1, pp. 111–123, 2003.

[5] P. Li, T. Zhang, B. Ma, "Unscented Kalman £lter for visual curve tracking," Image and Vision Computing, vol. 22, no. 2, pp. 157–164, 2004.

[6] B. Espiau, F. Chaumette, P. Rives, "A new approach to visual servoing in robotics," IEEE Trans. on Robotics and Automation, vol. 8, no. 3, pp. 313–326, June 1992.

[7] A. Blake, M. Isard, and D. Reynard, "Learning to track the visual motion of contours," Arti£cial Intelligence, vol. 78, pp. 101–134, 1995.

[8] F. Chaumette, "A £rst step toward visual servoing using image moments," In IEEE/RSJ Int Conf on Intelligent Robots and Systems, IROS'02, vol. 1, pp. 378–383, Lausanne, Switzerland, October 2002

[9] O. Tahri, F. Chaumette, "Application of moment invariants to visual servoing," In IEEE Int. Conf. on Robotics and Automation, ICRA'03, vol. 3, pp. 4276–4281, Taipeh, Taiwan, May 2003

[10] E. Malis, F. Chaumette and S. Boudet, "2 1/2D visual servoing," IEEE Trans. on Robotics and Automation, vol. 15, no.2, pp. 234–246, April 1999.

[11] G. Chesi, E. Malis and R. Cipolla, "Automatic segmentation and matching of planar contours for visual servoing," IEEE Int. Conf. on Robotics and Automation, ICRA'00, vol. 3, pp. 2753–2758, San Francisco, USA, April 2000.

[12] O. Faugeras, and F. Lustman, "Motion and structure from motion in a piecewise planar environment," Int. Journal of Pattern Recognition and Arti£cial Intelligence, vol. 2, no. 3, pp. 485–508, 1988.