Enhanced three-dimensional tracking using nonsingularity constraints

Huiying Chen
Youfu Li
City University of Hong Kong
Department of Manufacturing Engineering and Engineering Management
83 Tat Chee Avenue
Kowloon, Hong Kong
E-mail: velvet.chen@student.cityu.edu.hk

Abstract. We present a method for enhanced 3-D tracking using an interaction matrix. We developed two constraints to avoid the singularities and local minima of the interaction matrix. These constraints were combined with the minimization of tracking errors to identify suitable system configurations for enhanced tracking. Experiments were conducted to validate the method. © *2006 Society of Photo-Optical Instrumentation Engineers.* [DOI: 10.1117/1.2360519]

Subject terms: three-dimensional tracking; interaction matrix; nonsingularity constraint; singular value; condition number.

Paper 050896RRR received Nov. 15, 2005; revised manuscript received Mar. 5, 2006; accepted for publication Mar. 14, 2006; published online Oct. 18, 2006.

1 Introduction

The task of visual tracking is to continuously estimate and update the position and orientation of the target.[1] Depending on whether the depth information is used or not, visual tracking can be divided into three-dimensional (3-D) tracking and two-dimensional (2-D) tracking. The positions and orientations here refer to the *relative positions and orientations between the object and the vision system*. Thus, a tracking task can also be looked on from the point of view of dynamic view planning,[2] which aims at optimizing the viewpoints to estimate the object motion with minimum uncertainty.

In visual tracking, feature-based and model-based approaches can be found. Feature-based approaches track features such as geometrical primitives,[3] whereas model-based approaches use a model of the object.[1,4] In either case, tracking involves finding the corresponding feature points and using these points to obtain the object poses. The image feature locations may change due to motion of the vision system and/or the object, so that the relative position and orientation between the vision system and object play their role in tracking. The linkage between the change of the relative position and orientation and the change of features on the image plane is governed by a so-called *interaction matrix*.[5] The interaction matrix describes the projection of the 3-D velocity field onto the image motion field. The interaction matrix has been extensively studied in visual servoing.[6–8] Espiau et al.[8] proposed a method for computing the interaction matrix of any set of visual features defined on geometrical primitives, including points, lines, ellipses, cylinders, etc.

Because the interaction matrix provides the 3-D motion information and linkage between the 3-D object motion and the image motion, it is very important for visual tracking. However, because the interaction matrix suffers from singularities and local minima, it has mostly been computed at the "equilibrium," which is a possible solution to avoid singularities.[9] This calls for depth estimation. Recently, research on singularities of the interaction matrix has been conducted on singularity problems in visual servoing.[10] Michel and Rives[11] studied the problem of finding a particular visual feature set with three points where the interaction matrix had neither local minima nor singularities. Malis et al.[12] developed a triangular interaction matrix that had no singularity in the whole task space. Nevertheless, very little attention, if any, has been devoted to the singularity problem in visual tracking. Also, nonsingularity has rarely been explored in visual tracking as a constraint.

The singularity problem in visual servoing has been studied, for example, in an eye-in-hand system.[10] The purpose is to ensure that the control law for the camera motion is implementable. Generally, for a monocular eye-in-hand system, at least three visual feature points are needed to obtain the interaction matrix. In this case, the points' configuration is modified to avoid singularities.[11] This will be different in visual tracking, where the problem is whether the object location can be reliably recovered from the image features or not. Furthermore, instead of a subset of feature points, we study the singularities of every feature point for 3-D tracking.

In this paper, nonsingularity constraints on the interaction matrix are defined for 3-D tracking, to improve tracking performance. In the remainder of this paper, Sec. 2 presents the basic scheme of a 3-D tracking approach. In Sec. 3, the nonsingularity constraints are developed. Section 4 deals with the error analysis of 3-D tracking. In Sec. 5, the constraints for image feature location are defined. In Sec. 6 some experimental results are given.

2 3-D Tracking via an Interaction Matrix

2.1 *Visual Features and Relative Motion*

Let $\mathbf{p}(u,u)$ be a vector describing the current visual feature, where u and v are the image coordinates of the feature. The relative motion between the object and camera has the following relation to the motion of the visual feature:

$$\dot{\mathbf{p}} = \mathbf{L}\mathbf{T}_{\mathrm{re}}, \tag{1}$$

where $\mathbf{L}$ is the *interaction matrix*; $\mathbf{T}_{\mathrm{re}} = \begin{pmatrix} \mathbf{H} \\ \mathbf{\Omega} \end{pmatrix}$ is the relative

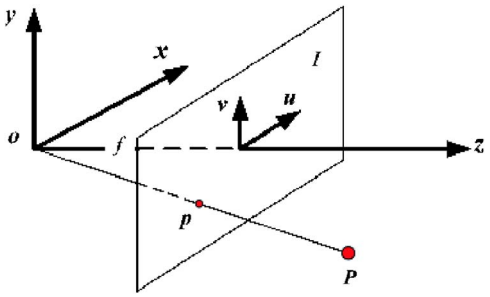

Fig. 1 Definition of the camera coordinate system.

motion between the object and the camera with $\mathbf{H}=(h_x,h_y,h_z)^T$, which is the translational velocity, and $\mathbf{\Omega}=(\omega_x,\omega_y,\omega_z)^T$, which is the angular velocity; and $\dot{\mathbf{p}}=(\dot{u},\dot{v})^T$ is the time derivative of the image feature due to the object relative motion $\mathbf{T}_{\text{re}}$. Therefore, the interaction matrix $\mathbf{L}$ is a 2×6 matrix.

Let $\mathbf{P}=(x,y,z)^T$ be a point on the object in the camera coordinate system. The camera coordinate system is defined as shown in Fig. 1, with the origin located at the optical center and the z axis being the optical axis. The relative motion between $\mathbf{P}$ and the camera can be described in vector notation as

$$\dot{\mathbf{P}}=\mathbf{\Omega}\times\mathbf{P}+\mathbf{H}=\mathbf{P}_{\times}\mathbf{\Omega}+\mathbf{H}, \tag{2}$$

where $\mathbf{P}_{\times}$ is a skew-symmetric matrix that concisely represents the cross product:

$$\mathbf{P}_{\times}=\begin{bmatrix}0 & -z & y\\ z & 0 & -x\\ -y & x & 0\end{bmatrix}. \tag{3}$$

If we assume that the object motion is rigid, then $\mathbf{\Omega}$ and $\mathbf{H}$ are the same for any $\mathbf{P}$. Equation (2) can be given as

$$\dot{x}=-z\omega_y+y\omega_z-h_x, \tag{4}$$

$$\dot{y}=-x\omega_z+z\omega_x-h_y, \tag{5}$$

$$\dot{z}=-y\omega_x+x\omega_y-h_z, \tag{6}$$

The change in the object's position can be obtained by integrating the relative motion:

$$\Delta\mathbf{P}=\int_0^{\delta t}\dot{\mathbf{P}}\,dt\approx\dot{\mathbf{P}}\,\delta t=(\mathbf{P}_{\times}\Omega+\mathbf{H})\,\delta t, \tag{7}$$

where δt is the sampling period of tracking.

Equation (1) yields

$$\mathbf{T}_{\text{re}}=\mathbf{L}^{+}\dot{\mathbf{p}}, \tag{8}$$

where $\mathbf{L}^{+}$ is the pseudo-inverse matrix of $\mathbf{L}$, so that $\mathbf{L}^{+}\mathbf{L}=\mathbf{I}$. Because the rank of $\mathbf{L}$ equals its number of rows. $\mathbf{L}^{+}$ can be obtained by

$$\mathbf{L}^{+}=\mathbf{L}^{T}(\mathbf{L}\mathbf{L}^{T})^{-1}. \tag{9}$$

According to Eq. (8), tracking can be achieved by searching for the corresponding image feature in a window and computing the feature motion $\dot{\mathbf{p}}$ to obtain the relative spatial motion $\mathbf{T}_{\text{re}}$.

2.2 *3-D Tracking with a Stereo Vision System*

Consider the canonical configuration of a stereo vision system, in which two cameras have the same focal length and their optical axes are parallel. The origins of the two camera coordinate systems are located at the centers of the image planes of the left and the right camera. The u axes are parallel to the baseline B, whereas the v axes are perpendicular to B. Assuming that a point $\mathbf{P}=(x,y,z)^T$ in the right camera frame is projected on the right image plane as a point $\mathbf{p}_r=(u_r,v_r)^T$ and on the left image plane as a point $\mathbf{p}_l=(u_l,v_l)^T$, we have

$$z=\frac{Bf}{u_r-u_l}, \tag{10}$$

where B is the baseline of the stereo system, and f is the focal length. The projection model of the right camera can be described as

$$x=u_r z/f, \tag{11}$$

$$y=v_r z/f. \tag{12}$$

From Eqs. (4) to (6), we can obtain the following equations for the position derivatives in the stereo system:

$$\dot{x}=-\frac{Bf}{u_r-u_l}\omega_y+\frac{v_r B}{u_r-u_l}\omega_z-h_x, \tag{13}$$

$$\dot{y}=-\frac{u_r B}{u_r-u_l}\omega_z+\frac{Bf}{u_r-u_l}\omega_y-h_y, \tag{14}$$

$$\dot{z}=-\frac{B}{u_r-u_l}(v_r\omega_x-u_r\omega_y)-h_z, \tag{15}$$

where $\omega_x,\omega_y,\omega_z$ and h_x,h_y,h_z are the rotation velocities and translation velocities of point P, respectively. Substituting Eqs. (13) and (15) into Eqs. (11) and (12), and using the quotient rule, we have

$$\dot{u}_r=f\frac{z\dot{x}-x\dot{z}}{z^2}=-\frac{u_r-u_l}{B}h_x+\frac{(u_r-u_l)}{Bf}h_z+\frac{u_r v_r}{f}\omega_x-\frac{f^2+u_r^2}{f}\omega_y+v_r\omega_z. \tag{16}$$

Similarly,

$$\dot{v}_r=-\frac{u_r-u_l}{B}h_y+\frac{(u_r-u_l)v_r}{Bf}h_z+\frac{f^2+v_r^2}{f}\omega_x-\frac{u_r v_r}{f}\omega_y+u_r\omega_z. \tag{17}$$

Finally, the last two equations can be rewritten in a matrix form as

$$\begin{bmatrix} \dot{u}_r \\ \dot{v}_r \end{bmatrix} = \begin{bmatrix} -\frac{u_r-u_l}{B} & 0 & \frac{(u_r-u_l)u_r}{Bf} & \frac{u_r v_r}{f} & -\frac{f^2+u_r^2}{f} & v_r \\ 0 & -\frac{u_r-u_l}{B} & \frac{(u_r-u_l)v_r}{Bf} & \frac{f^2+v_r^2}{f} & -\frac{u_r+v_r}{f} & -u_r \end{bmatrix} \times \begin{bmatrix} h_x \\ h_y \\ h_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \tag{18}$$

which leads to the interaction matrix for the stereo systems:

$$\mathbf{L}_s = \begin{bmatrix} -\frac{u_r-u_l}{B} & 0 & \frac{(u_r-u_l)u_r}{Bf} & \frac{u_r v_r}{f} & -\frac{f^2+u_r^2}{f} & v_r \\ 0 & -\frac{u_r-u_l}{B} & \frac{(u_r-u_l)v_r}{Bf} & \frac{f^2+v_r^2}{f} & -\frac{u_r v_r}{f} & -u_r \end{bmatrix}, \tag{19}$$

or

$$\mathbf{L}_s = \begin{bmatrix} -\frac{f}{z} & 0 & \frac{u_r}{z} & \frac{u_r v_r}{f} & -\frac{f^2+u_r^2}{f} & v_r \\ 0 & -\frac{f}{z} & \frac{v_r}{z} & \frac{f^2+v_r^2}{f} & -\frac{u_r v_r}{f} & -u_r \end{bmatrix}, \tag{20}$$

so that

$$\dot{\mathbf{p}}_r = \begin{pmatrix} \dot{u}_r \\ \dot{v}_r \end{pmatrix} = \mathbf{L}_s \mathbf{T}_{\mathrm{re}}. \tag{21}$$

3 Nonsingularity Constraints in the Interaction Matrix

Because the interaction matrix is used to obtain the object's 3-D position and orientation, its properties affect tracking performance. The vision system will fail in tracking if the interaction matrix falls into singularities. To avoid this, two constraints—on the *smallest singular value* and the *condition number*—are employed.

3.1 *The Smallest Singular Value*

The singular-value decomposition is based on decomposing a matrix into two matrices $\mathbf{U}_\Sigma$ and $\mathbf{V}_\Sigma$ and a diagonal matrix $\mathbf{\Sigma}$, containing scale factors called singular values. This decomposition of an interaction matrix $\mathbf{L}$ is expressed as

$$\mathbf{L}_{(2\times 6)} = \mathbf{U}_\Sigma \mathbf{\Sigma} \mathbf{V}_\Sigma^T = \mathbf{U}_{\Sigma(2\times 2)} \underbrace{\begin{pmatrix} \sigma_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 & 0 & 0 \end{pmatrix}}_{\mathbf{\Sigma}} \mathbf{V}_{\Sigma(6\times 6)}^T. \tag{22}$$

Here $\mathbf{U}_\Sigma$ is a basis of the two-parameter 2-D image-motion space, and $\mathbf{V}_\Sigma$ is a basis of the six-parameter 3-D spatial-motion space. Each singular value (σ_1 or σ_2) in $\mathbf{\Sigma}$ corresponds to the projection on a single image dimension of a column in $\mathbf{U}_{\Sigma(2\times 2)}$ for image feature change and a row in $\mathbf{V}_{\Sigma(6\times 6)}$ for 3-D spatial change. The image-feature motion projected on the scene is the sum of all partial projections scaled by the corresponding singular values in $\mathbf{\Sigma}$.

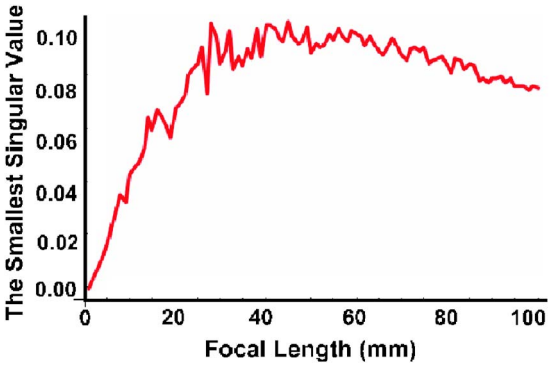

Fig. 2 Influence of the focal length on the smallest singular value when z=1000 mm, B=200 mm, u=v=100 pixels.

In the singular-value decomposition, we always define that $\sigma_1 \geq \sigma_2 \geq 0$. If the interaction matrix is singular, which means that at least one of its singular values is zero, the vision system will lose the projection information in certain direction(s). This will then cause 3-D tracking failure. Therefore we impose a constraint on the smallest singular value (σ_2 here) for 3-D tracking. This constraint indicates the worst informative direction of visual projection and provides a lower bound for this direction. The constraint is defined as

$$G_1{:}\{\sigma_2 \in \sigma | \sigma - \xi > 0\}, \tag{23}$$

where ξ is a positive lower bound of the smallest singular value.

Because the smallest singular value cannot have an analytic solution, we used simulations to examine its properties. Our study shows that the focal length of the vision system, the depth and feature location can affect the smallest singular value (σ_2).

In Fig. 2, the mean values of σ_2 are plotted for different values of the focal length. It is seen that there is a global maximum of σ_2, which corresponds to the best nonsingular condition of the interaction matrix. Then the constraint on σ_2 can be written as a constraint on the focal length,

$$G_{1f}{:}\{f_{\min}|_\xi < f < f_{\max}|_\xi\}, \tag{24}$$

where $f_{\min}$ and $f_{\max}$ are the lower and upper bound for the focal length corresponding to ξ.

Similarly, according to the simulation result in Fig. 3, the constraint on σ_2 also depends on the depth value z:

$$G_{1z}{:}\{z > z_{\min}|_\xi\}, \tag{25}$$

where $z_{\min}$ is the lower bound for the depth corresponding to ξ.

The result in Fig. 4 shows that the image feature's location can also affect the smallest singular value. Features located near to the image center have larger singular values. Features located far away from the image center tend to give smaller singular values, which may cause singularity problems.

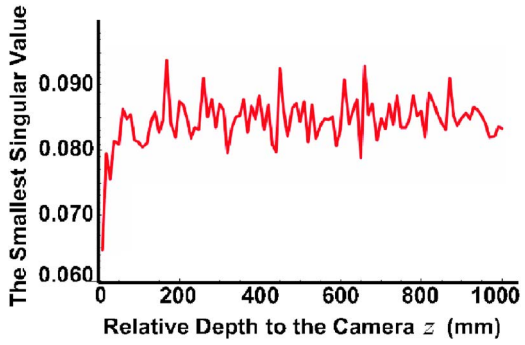

Fig. 3 Influence of depth value on the smallest singular value when f=30 mm, B=200 mm, $u=v$=100 pixels.

Therefore, the nonsingularity constraint on image feature location is introduced as

$$G_{1r}:\{r < r_{\max}|_{\xi}\}, \tag{26}$$

where r is the distance from the image feature point to the image center, and $r_{\max}$ is the maximum distance with respect to ξ.

3.2 *Condition Number*

As mentioned in the previous subsection, the constraint on σ_2 reveals the worst informative direction. Now the condition number $\kappa(\mathbf{L})$ is introduced to give a *balance constraint* among different informative directions defined by the interaction matrix. $\kappa(\mathbf{L})$ is defined as the ratio of the larger to the smaller singular value of the interaction matrix $\mathbf{L}$:

$$\kappa(\mathbf{L}) = \frac{\sigma_1}{\sigma_2}. \tag{27}$$

A system is said to be singular if the condition number is infinite. In such a case, the system will be ill conditioned, which may lead to large computational errors and local minima. The value of $\kappa(\mathbf{L})$ can be calculated from the matrix norm as follows:

$$\kappa(\mathbf{L}) = \|\mathbf{L}\|\|\mathbf{L}^{+}\|. \tag{28}$$

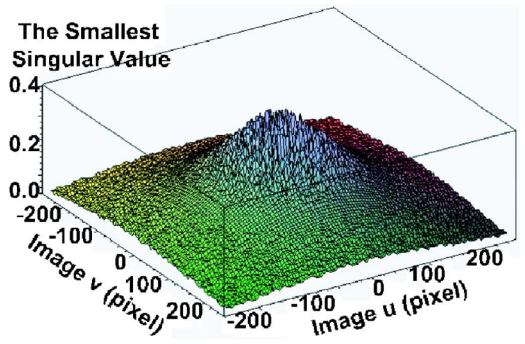

Fig. 4 Influence of image feature location on the smallest singular value when z=1000 mm, B=200 mm, f=30 mm.

For a stereo vision system, substituting Eq. (20) into Eq. (9), we can calculate the pseudo-inverse matrix of the interaction matrix:

$$\mathbf{L}_s^{+} = \eta \begin{bmatrix} -fz(v_r^2+f^2) & fzv_ru_r \\ fzv_ru_r & -fz(u_r^2+f^2) \\ f^2zu_r & f^2zv_r \\ 0 & z^2f(v_r^2+u_r^2+f^2) \\ -z^2f(v_r^2+u_r^2+f^2) & 0 \\ z^2v_r(v_r^2+u_r^2+f^2) & -z^2u_r(v_r^2+u_r^2+f^2) \end{bmatrix}, \tag{29}$$

where $\eta = 1/[(v_r^2+u_r^2+f^2)(f^2+z^2v_r^2+f^2z^2+u_r^2z^2)]$. Substituting Eq. (29) into Eq. (28) and reducing the results, we can obtain a simple form of the condition number as

$$\kappa(\mathbf{L}_s) = \frac{u_r^2+v_r^2+f^2}{f^2}. \tag{30}$$

Equation (30) can also be written as

$$\kappa(r,f) = \frac{r^2+f^2}{f^2}. \tag{31}$$

Equation (31) indicates that the condition number of the interaction matrix is only affected by the focal length and the location of the image feature point.

Consequently, the nonsingularity constraint on the condition number can be expressed as

$$G_2 : \left\{ \frac{r^2+f^2}{f^2} = 1 + \frac{r^2}{f^2} < \kappa_0 \right\}, \tag{32}$$

where κ_0 is a threshold for the condition number. This constraint can be decomposed into two constraints on f and r as

$$G_{2f}:\{f > f_{\min}|_{\kappa_0,r}\}, \tag{33}$$

$$G_{2r}:\{r < r_{\max}|_{\kappa_0,f}\}. \tag{34}$$

3.3 *Summary of Nonsingularity Constraints*

When we take both the smallest singular value and the condition number into consideration, the combined nonsingularity constraints on focal length, image feature location, and depth location can be given as

$$G_f:\{f_{\min}|_{\xi} < f < f_{\max}|_{\xi}\} \cap \{f > f_{\min}|_{\kappa_0,r}\}, \tag{35}$$

$$G_r:\{r < r_{\max}|_{\xi}\} \cap \{r < r_{\max}|_{\kappa_0,f}\}, \tag{36}$$

$$G_z:\{z > z_{\min}|_{\xi}\}. \tag{37}$$

3-D tracking can be taken as the process of dynamic view planning to continuously obtain the 3-D viewpoints of the vision system. In such a process, nonsingularity constraints can be imposed on the system configuration to provide enhanced tracking performance.

According to the discussions in the previous subsections, we are now able to impose nonsingularity constraints (G_f, G_r, and G_z) on the focal length, image feature location, and depth location. Once the focal length and depth location are determined, constraints on the image feature location can be transformed to constraints on the relative position between the vision system and the object:

$$\begin{pmatrix} u_g \\ v_g \\ 1 \end{pmatrix} = \mathbf{M}(f_g) \begin{pmatrix} x \\ y \\ z_g \\ 1 \end{pmatrix}. \tag{38}$$

Thus

$$\begin{pmatrix} x \\ y \\ z_g \\ 1 \end{pmatrix} = \mathbf{M}^{+}(f_g) \begin{pmatrix} u_g \\ v_g \\ 1 \end{pmatrix}, \tag{39}$$

where (u_g, v_g) is the constrained image location, (x, y, z) is the relative 3-D position, f_g and z_g are the constrained focal length value and depth value, and $\mathbf{M}(f_g)$ and $\mathbf{M}^{+}(f_g)$ are the projection matrix and its pseudo-inverse. After the transformation (39), the nonsingularity constraints can be imposed on the system configuration, namely, the relative 3-D position and the focal length.

In a robot vision system where the viewpoint and system configuration can be controlled actively, the nonsingularity constraints can be implemented online for 3-D tracking. When a system is not reconfigurable, the constraints may be implemented offline. System configurations with satisfactory levels for the nonsingularity constraints can be simulated in advance and saved in a database. That database then can be accessed during the tracking process. According to the system configuration (focal length, relative position, and orientation) at certain times during the tracking, a satisfaction level for nonsingularity constraints in that configuration can be obtained by using the database. Once the level is down to a predetermined threshold, the system can immediately provide a warning of impending tracking failure.

4 Error Analysis

4.1 *Modeling of 3-D Tracking Error*

We denote the observed feature motion with errors as $\dot{\mathbf{p}}'$. The errors may include the system error of the vision system, quantization error of the camera, segmentation error, approximation error of the first derivative, sensing delay error, and so on. We have

$$\dot{\mathbf{p}}' = \dot{\mathbf{p}} + \Delta\dot{\mathbf{p}}, \tag{40}$$

where $\Delta\dot{\mathbf{p}}$ is the feature motion error with $\Delta\dot{\mathbf{p}} = (\Delta\dot{u}, \Delta\dot{v})^T$. Without loss of generality, let

$$\begin{pmatrix} \Delta\dot{u} \\ \Delta\dot{v} \end{pmatrix} = \mathbf{Q}_{2\times 2} \begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix},$$

where $\mathbf{Q}$ is a scale matrix. When $\Delta\dot{\mathbf{p}}$ is projected to the object motion field, from Eq. (8) we have the estimated object motion error as

$$\Delta\mathbf{T}_{\mathrm{re}} = \mathbf{L}_s^{+}\,\Delta\dot{\mathbf{p}} + \Delta\mathbf{L}_s^{+}\,\dot{\mathbf{p}} = (\mathbf{L}_s^{+}\mathbf{Q} + \Delta\mathbf{L}_s^{+})\dot{\mathbf{p}}, \tag{41}$$

where $\Delta\mathbf{T}_{\mathrm{re}} = \begin{pmatrix} \Delta\mathbf{H} \\ \Delta\boldsymbol{\Omega} \end{pmatrix}$ and $\Delta\mathbf{L}_s^{+}$ is the error of $\mathbf{L}_s^{+}$.

If quantization error is considered, we have

$$\Delta\mathbf{L}_s^{+} = \frac{\partial\mathbf{L}_s^{+}}{\partial z_s}\Delta z + \frac{\partial\mathbf{L}_s^{+}}{\partial u}\Delta u_r + \frac{\partial\mathbf{L}_s^{+}}{\partial v_r}\Delta v_r, \tag{42}$$

where Δz is the depth estimation error, and Δu_r and Δv_r are the quantization errors along the u and v image axes of the right camera, respectively. We assume that Δu_r and Δv_r are independent of each other.

According to the principles of error propagation, Eq. (10) leads to

$$\Delta z = \frac{-Bf}{(u_r - u_l)^2}\Delta(u_r - u_l), \tag{43}$$

where $\Delta(u_r - u_l)$ is the disparity error. Because Δu_r and Δu_l, are independent of each other, if we assume that the two cameras have the same quantization error, then we have $\Delta(u_r - u_l) = 2\,\Delta u_r$. Equation (43) yields

$$\Delta z = \frac{-2Bf}{(u_r - u_l)^2}\Delta u_r = \frac{-2z^2}{Bf}\Delta u_r. \tag{44}$$

Thus (42) becomes

$$\Delta\mathbf{L}_s^{+} = \frac{\partial\mathbf{L}_s^{+}}{\partial z}\frac{-2z^2}{Bf}\Delta u_r + \frac{\partial\mathbf{L}_s^{+}}{\partial u_r}\Delta u_r + \frac{\partial\mathbf{L}_s^{+}}{\partial v_r}\Delta v_r. \tag{45}$$

Here $\mathbf{L}_s^{+}$ is a 6×2 matrix. We denote the two component matrices of $\mathbf{L}_s^{+}$ as $\mathbf{L}_{s1}^{+}$ and $\mathbf{L}_{s2}^{+}$, and those of $\Delta\mathbf{L}_s^{+}$ as $\Delta\mathbf{L}_{s1}^{+}$ and $\Delta\mathbf{L}_{s2}^{+}$, namely,

$$\mathbf{L}_s^{+} = \begin{pmatrix} \Delta\mathbf{L}_{s1(3\times 2)}^{+} \\ \Delta\mathbf{L}_{s2(3\times 2)}^{+} \end{pmatrix}, \quad \Delta\mathbf{L}_s^{+} = \begin{pmatrix} \Delta\mathbf{L}_{s1(3\times 2)}^{+} \\ \Delta\mathbf{L}_{s2(3\times 2)}^{+} \end{pmatrix}. \tag{46}$$

Following the notation of Eqs. (2) and (41), we have

$$\Delta\mathbf{H} = \mathbf{L}_{s1}^{+}\,\Delta\dot{\mathbf{p}} + \Delta\mathbf{L}_{s1}^{+}\,\dot{\mathbf{p}} = (\mathbf{L}_{s1}^{+}\mathbf{Q} + \Delta\mathbf{L}_{s1}^{+})\cdot\dot{\mathbf{p}}, \tag{47}$$

$$\Delta\boldsymbol{\Omega} = \mathbf{L}_{s2}^{+}\,\Delta\dot{\mathbf{p}} + \Delta\mathbf{L}_{s2}^{+}\,\dot{\mathbf{p}} = (\mathbf{L}_{s2}^{+}\mathbf{Q} + \Delta\mathbf{L}_{s2}^{+})\cdot\dot{\mathbf{p}}. \tag{48}$$

According to Eq. (2), the estimated motion error in the 3-D object space can be obtained as

$$\Delta\dot{\mathbf{P}} = [\mathbf{P}_x]\,\Delta\boldsymbol{\Omega} + \Delta\mathbf{H}, \tag{49}$$

where $\Delta\dot{\mathbf{P}} = (\Delta\dot{P}_x, \Delta\dot{P}_y, \Delta\dot{P}_z)^T$. Consequently, similar to Eq. (7), the position error of tracking is

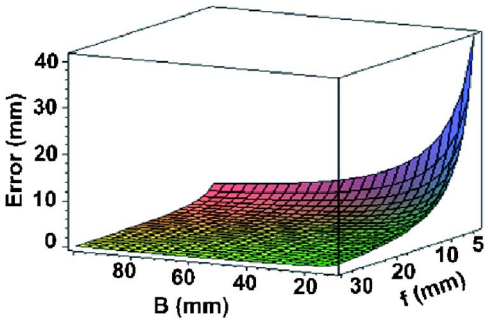

Fig. 5 Tracking error ε_P as a function of B and f when $z=1000$ mm, $u=v=100$ pixels.

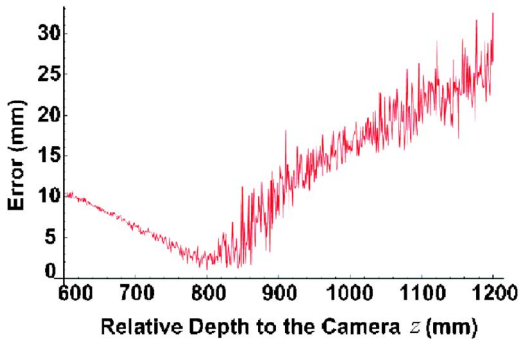

Fig. 6 Best-focus location for minimizing tracking errors when $f=30$ mm, $B=200$ mm, $u=v=100$ pixels.

$$\varepsilon_P = \|\Delta \mathbf{P}\| \approx \|\Delta \dot{\mathbf{P}}\| \delta t = (\Delta \dot{P}_x^2 + \Delta \dot{P}_y^2 + \Delta \dot{P}_z^2)^{1/2} \delta t, \tag{50}$$

where δt is the sampling period of tracking. The tracking error ε_P can also be regarded as the tracking uncertainty, and it is a function of time.

4.2 *Influence of System Configuration on Tracking Errors*

Following the definition (50), the tracking error ε_P is used as the cost function of the tracking system. Then the parameters for optimal tracking can be obtained by minimization of ε_P.

4.2.1 *Focal length and baseline*

According to our simulation study, both the baseline value B and the focal length value f can affect the tracking error. As shown in Fig. 5, the tracking error is approximately inversely proportional to B and f. This result tallies with what is observed in the real camera setup. According to the projection rule, a large focal length f can provide better sensing over a wide range. As to the baseline value B, following the triangulation rule, the larger the baseline, the more sensitive the depth estimation is. Thus a larger B value results in smaller tracking errors.

4.2.2 *Best-focus location*

3-D tracking is different from 2-D tracking in that it uses depth information, which can help improve the tracking performance. Our simulation study on the average errors at different depth (z) locations shows that there is an optimal z location at which the tracking error reaches its global minimum as shown in Fig. 6. This is because the tracking resolution is inversely proportional to z, which indicates that a large z is desirable for better tracking. However, when z becomes large, the vision system becomes less sensitive to the object motion. Therefore, the optimal z location is based on the best compromise between the positioning uncertainty and the sensitivity of the vision system. This location in fact corresponds to the best-focus location (BFL) studied in our previous research.[13]

5 Combined Constraint for Image Feature Location

When the vision system is reconfigurable, we can try to keep the feature point at a specific position (u^*, v^*) on the image to minimize the tracking error and improve the tracking performance. Assume that the image feature velocities at a certain sampling instant are $\dot{u}_i, \dot{v}_i$. Let $\alpha_i = \dot{u}_i / \dot{v}_i$, where α_i is a scale factor. We have examined how different image feature locations affect the tracking errors. As shown in Fig. 7, the distribution of tracking errors on the image plane is centrosymmetric, and it converges to two global minima along the line $u=\alpha_i v$ on the edge of the image plane. Figure 7 also shows how the tracking errors change with α_i.

According to the discussion in Sec. 3, in order to satisfy nonsingularity constraints, image feature points should be located close to the image center. However, to minimize tracking error, image feature points should be located away from the image center on the line $u=\alpha_i v$. Therefore, an optimal targeting position can only be achieved by making a compromise between the satisfaction of nonsingularity constraints and the minimization of the tracking error. Figure 8 shows the region of such positions. Note that the optimal targeting position is not a fixed location on the image plane. Different relative motions between the camera and the object or different camera configurations and setups will lead to different optimal targeting positions.

From the preceding analysis, the optimal targeting position is the image point location at which the tracking error is minimized. Thus, in practical applications, we can adjust the vision system's configuration to maintain the observed feature point at this position during tracking.

6 Experimental Results

6.1 *System Setup*

The implementation of the proposed tracking method was conducted using our vision system, with a PC-based IMPCI system and a variable-scan framegrabber. This system supports many real-time processing functions, including some feature extraction such as edge detection. Our algorithms were developed in the VC++ programming language and run as imported functions by ITEX-CM. The system setup consists of two identical cameras (Pulnix model TM-765), with a resolution of 768×582 pixels. The two cameras were separated by a baseline of 190.2 nm. Each camera was calibrated separately in advance to evaluate their internal parameters. Also, the pair of cameras was calibrated to obtain the external parameters of the stereo system.

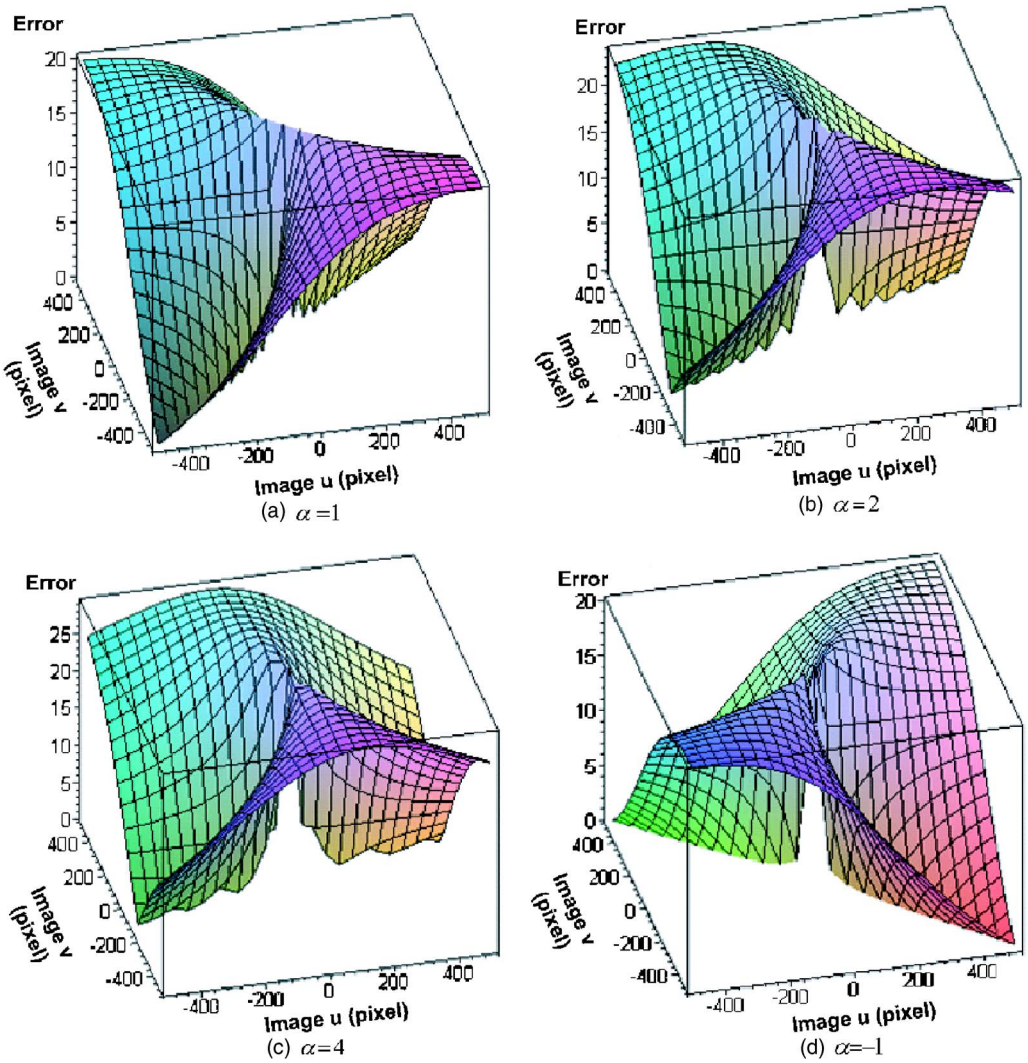

Fig. 7 Tracking error converges on line $u=\alpha v$ at different α when $z=1000$ mm, $B=200$ mm, $f=30$ mm.

6.2 *Tracking Implementation*

We defined the tracking error as the distance between the estimated object location and the true object location at each sampling instant. A basic experiment was conducted to reveal the relation between the tracking error and the object velocity. In this experiment, 100 samples were taken

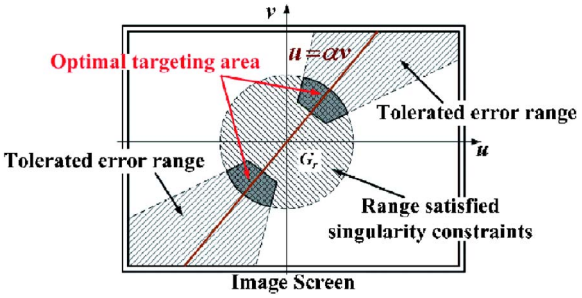

Fig. 8 The optimal targeting area on the image plane.

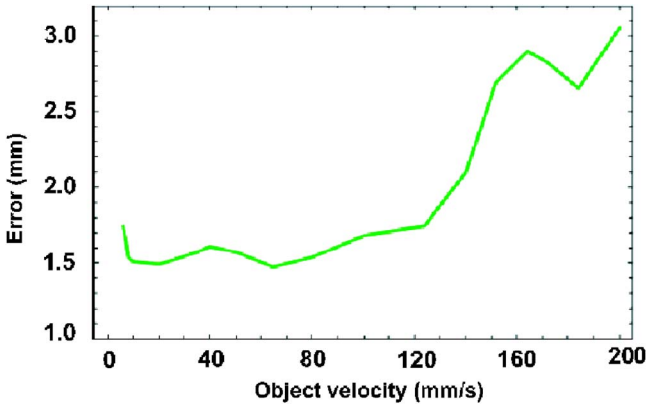

Fig. 9 Tracking error versus object velocity when $z=1050$ mm, $B=190.2$ mm, $f=28$ mm.

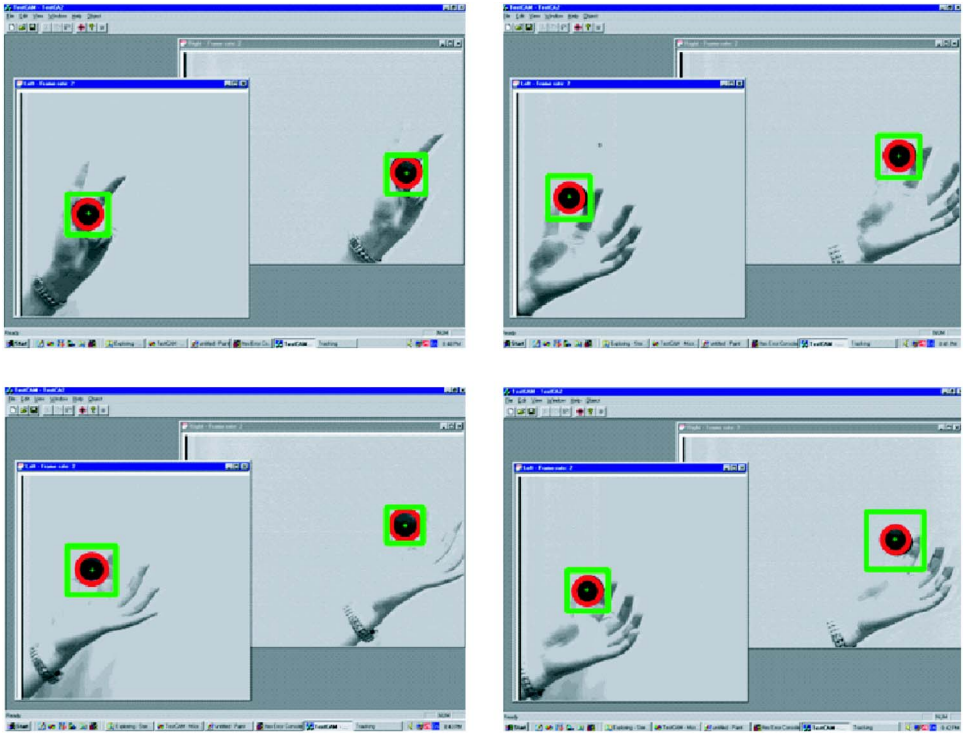

Fig. 10 Stereo tracking by our method.

and used for quantitative comparison. The mean tracking errors at different object velocities are shown in Fig. 9.

The tracking errors tend to increase when the object moves faster, as seen in Fig. 9. Therefore, we have developed a tracking method based on the use of an adaptive search window whose size is proportional to the object velocity. This method enables us to efficiently and reliably search for a target with changing velocities within a small area of the predicted window. Assuming that the feature location $\mathbf{p}_t$ at time t on the right camera image plane, the relative motion $\mathbf{T}_{\mathrm{re}}(t)$, and the interframe time τ are given, then the predicted location (center location) of the search window can be estimated as

$$\mathbf{p}_{t+\tau} = \mathbf{p}_t + \int^{t+\tau} \dot{\mathbf{p}}_t \, \mathrm{d}t \approx \mathbf{p}_t + \dot{\mathbf{p}}_t \tau = \mathbf{p}_t + \tau \mathbf{L}_s(t) \mathbf{T}_{\mathrm{re}}(t). \tag{51}$$

To reduce the searching time, the window size is here made adaptable with respect to tracking uncertainty (estimated tracking errors). The change of the window size is defined as proportional to the product of the image velocity and tracking uncertainty as follows:

$$\mathbf{s}_w(t) = \begin{pmatrix} \delta_u(t) \\ \delta_v(t) \end{pmatrix} = \begin{pmatrix} \delta_{u0} \\ \delta_{v0} \end{pmatrix} + \lambda \varepsilon_P(t) \begin{pmatrix} |\dot{u}(t)| \\ |\dot{v}(t)| \end{pmatrix}, \tag{52}$$

where λ is a positive scale factor, $\varepsilon_P(t)$ is the estimated tracking error from Eq. (50), $(\delta_u(t), \delta_v(t))^T$ contains the window sizes along the u and v image coordinates, respectively, and $(\delta_{u0}, \delta_{v0})^T$ gives the minimum window size. Similarly, the adaptive search window for the left camera can be derived.

6.3 *Tracking in 3-D*

Firstly, we implemented our method to track a ball. The location of the search window for the next step was predicted using the position and velocity information currently available. The size of the search window was changed according to the predicted tracking error and the object velocity. Some examples of snapshots in the tracking are shown in Fig. 10. Here the tracking is formulated to take into account the depth information in addition to the position of the ball in the image.

In another experiment, we implemented our method in tracking a moving hand as shown in Fig. 11. In this case, the 3-D orientation as well as the 3-D position of the hand has to be considered in the tracking. The adaptive search window was also adopted, although it is not displayed in Fig. 11 for simplicity's sake. A tracking rate of about 10 frames/s was achieved in the implementation.

6.4 *Nonsingularity Constraints for Tracking*

As discussed in Sec. 3, the singularity properties of the interaction matrix can be affected by three parameters, image feature location (distance to the image center), focal length, and depth location. Since singularity can further affect tracking performance, we examined the influence of those parameters on the tracking errors and verified their effects on singularity.

In order to separate the influence on tracking error of one parameter from the others, in the experiments for examining a certain parameter we fixed the other two parameters at their *typical values*. Those typical values were obtained empirically in advance, which made them satisfy the following two criteria: (1) they are representative of the

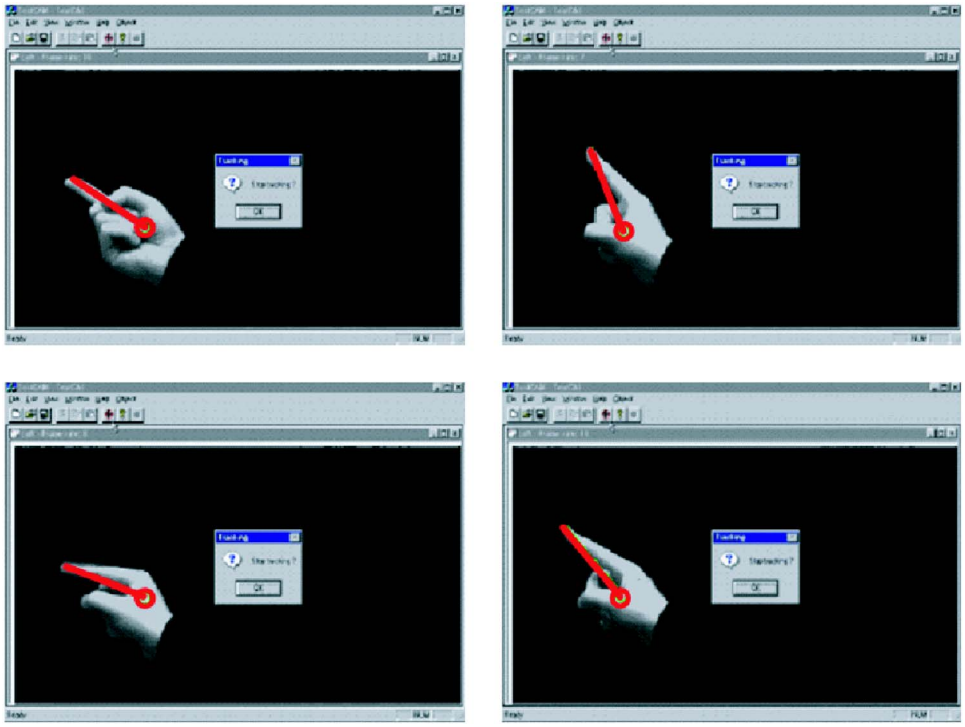

Fig. 11 Tracking a hand.

general case; and (2) at these values, the system will have relatively small tracking errors.

6.4.1 *Image feature location*

The depth location was fixed at a typical value of 1050 nm and the focal length at 28 nm for this experiment. As discussed in Sec. 5, image feature location itself can also affect tracking errors. Thus, a special motion pattern is adopted to eliminate such influence. We used a point object for the experiment and made the object undergo uniform circular motion around the center with different diameters (see Fig. 12). Using this motion, at any point on the circle, the magnitude of velocity is uniform and the direction of velocity is perpendicular to the line drawn from the object to the center of the circle. If the influence of singularity is not considered, then according to Fig. 7 an object on the same circle will have the same effect on the tracking errors. To eliminate the influence of velocity on the tracking error (as shown in Fig. 9) the object was made to move along different circles at the same speed. This made it possible to extract the relatively pure influence of singularity on the tracking errors.

The tracking error was defined as the spatial distance between the estimated value of the object location and its true value (see Fig. 13).

The object was moved along various concentric circles repeatedly, and the average tracking errors were obtained at each sampling time (see Fig. 14). The obtained result in Fig. 14 was then analyzed. The mean tracking errors on different concentric circles were calculated and are given in Table 1.

As shown in Fig. 15, the absolute tracking errors are affected by image feature location r. Note that the errors

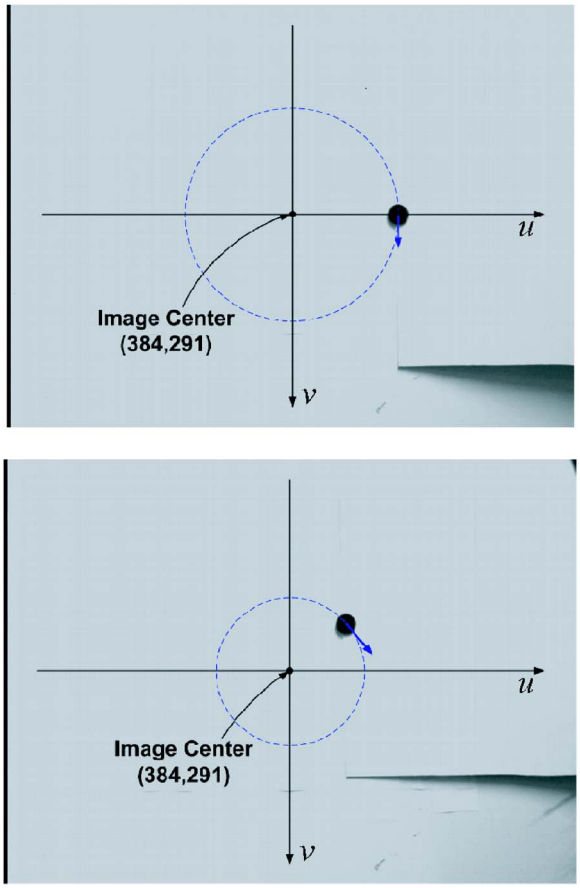

Fig. 12 Uniform circular motion at different diameters.

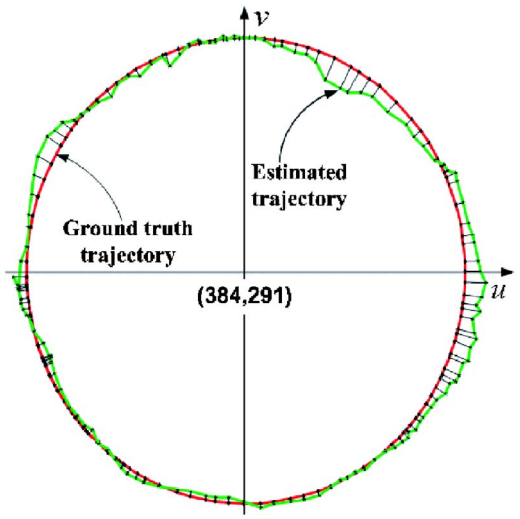

Fig. 13 Tracking errors on a circle.

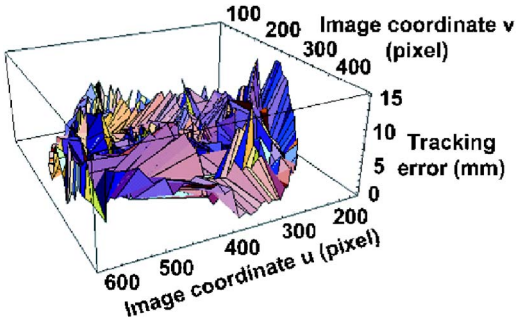

Fig. 14 Average tracking errors for different image locations.

Table 1 Data analysis on different concentric circles.

Test diameter $2r$ (pixels)	Mean tracking error (mm)	Standard deviation (mm)	No. of samples
400	13.7	15.6	776
360	−12.1	15.0	539
320	−10.1	16.6	701
280	9.6	15.8	614
240	11.5	14.5	557
200	5.2	13.5	712
160	−4.6	11.1	615
120	5.9	7.6	520
80	−5.5	8.6	474

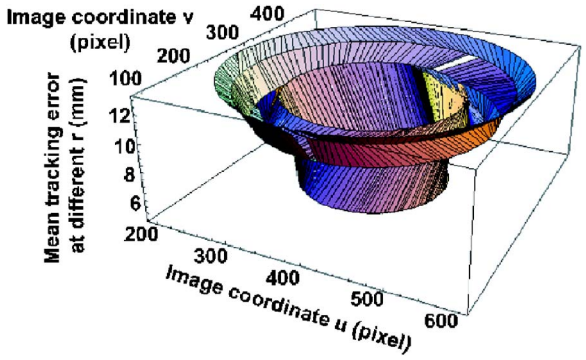

Fig. 15 Absolute tracking errors versus radius *r*.

caused by lens distortion may contribute to the tracking errors here. Therefore, we conducted separate experimental examinations on the static errors, which include the effect of the lens distortion in the vision system. With our camera calibration, the mean static errors were found to be within 1.0 mm for the setup. Thus, the influence of the lens distortion on the tracking errors can be ignored. This experimental result is consistent with the simulation result shown in Fig. 4, which verified the effect of singularity on tracking errors. When the image feature point is close to the center, the smallest singular value becomes large, so that better tracking performance can be achieved.

When a threshold for tracking error is defined, the corresponding nonsingularity constraint on r can be determined empirically. As shown in Table 1, there is a large variation in tracking error when the diameter changes from 240 to 200 pixels. This suggests that a diameter of 200 pixels can be used as the threshold $r_{\max}|_{\xi}$ for the nonsingularity constraint.

6.4.2 *Focal length*

In this experiment, the depth was fixed at 1050 mm and the image feature was fixed on a circle of 160-pixel diameter. The mean tracking errors with their variances at different focal lengths are shown in Fig. 16, which shows that there is an optimal focal length value that minimizes the tracking error. This is consistent with the result of Fig. 2 in Sec. 3, suggesting that there is a global maximum in the smallest singular value at which its corresponding focal length best satisfies the nonsingularity constraint G_{1f}.

However, the real situation is far more complicated. On one hand, the focal length affects the condition number

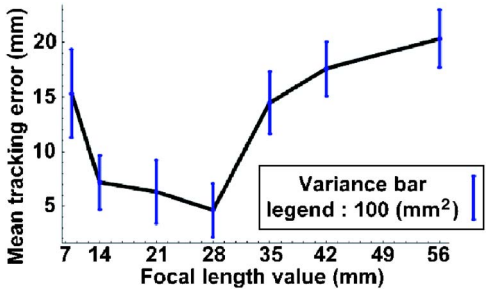

Fig. 16 Influence of focal length on tracking errors.

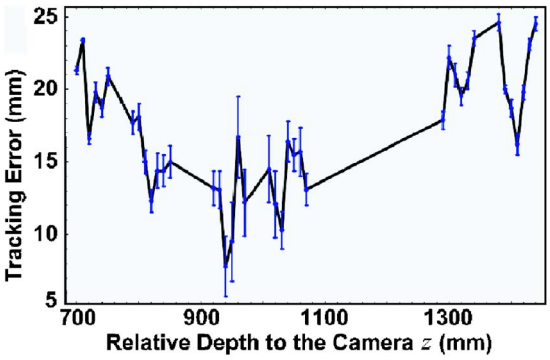

Fig. 17 Effect of z on tracking errors when $B=190.2$ mm, $f=28$ mm.

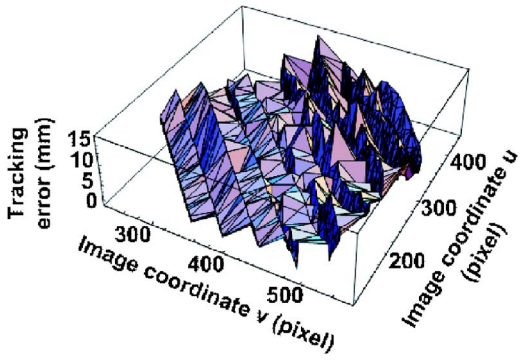

Fig. 19 Average tracking errors for different image locations.

(G_{2f} in Sec. 3), so that a large focal length can provide a small condition number and better nonsingularity. Thus, a large focal length is desired to reduce the tracking errors due to singularity. On the other hand, the focal length itself can directly affect tracking errors (see Fig. 5 in Sec. 4). When the focal length is too large or too small ($f<10$ mm or $f>35$ mm in this case), the blurring effects of the target will affect the precision of tracking adversely. The result in Fig. 16 therefore is considered as a combination of these effects.

6.4.3 *Depth location*

As shown in the simulation result in Fig. 3, the vision system will suffer from singularity problems only when the depth z is close to zero. In a real experimental setup, when z is very small, the blurring will make it unsuitable for tracking. We thus ignored the nonsingularity constraint. Instead, an in-focus (blurring) constraint can be imposed on z.

6.5 *Best-Focus Location*

In a tracking task, the z value can affect tracking errors as discussed in Sec. 4. In our implementation, the search window size depends not only on the object velocity but also on the value of z. Experiments were conducted to identify the relationship between the tracking errors and z. In these tests, the object moved towards and away from the stereo system in different z locations. When the predicted 3-D positions of the object were compared with their true values, the mean tracking errors with their variances at different z locations could be obtained as shown in Fig. 17. It can be seen that there is a BFL that minimizes the tracking errors. The experimental results show that there is a global minimum in the tracking error at $z\approx 940$ mm.

6.6 *Optimal Targeting Area*

The simulation result in Fig. 7 in Sec. 5 indicates that the tracking error converges along the line $u=\alpha v$ away from the image center. Also, if the influence of singularity is considered, there should be an optimal targeting area on the image plane that provides better tracking performance (see Fig. 8 in Sec. 5). Experiments have been conducted to identify this area. We made a point object undergo the same uniform straight-line motion along different parallel lines $u_i=-v_i+b_i$, with $i=1,\ldots,9$ and $b_5=0$ (see Fig. 18). The direction of the velocity of the object was restricted to $\dot{u}=-1\cdot\dot{v}$, which means that $\alpha=-1$. Then according to the simulation result in Fig. 7, the tracking errors are expected to converge on the line $u=-v$ (L5 in Fig. 18). The object moved along each of the parallel lines repeatedly, and the average tracking errors were obtained at each sampling time (see Fig. 19). The mean tracking errors on different lines are shown in Fig. 20. The smallest mean tracking error appears on L5. In other words, the tracking errors converge along the line $u=-v$. It should be noted that in Fig. 7, there is a global maximum in the tracking error at the image center. However, in that simulation the nonsingularity constraint was not taken into account. In experi-

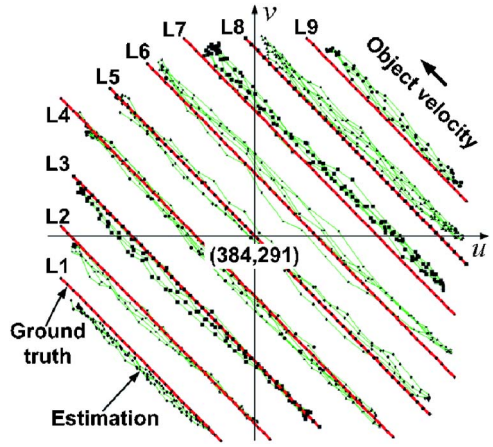

Fig. 18 Object trajectory in the image.

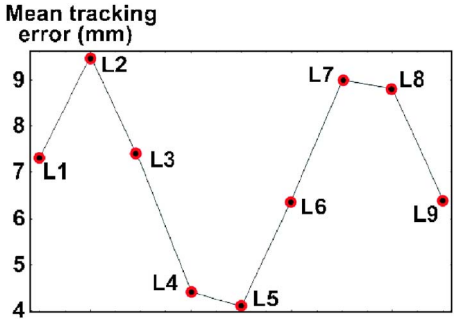

Fig. 20 Absolute tracking errors on different lines.

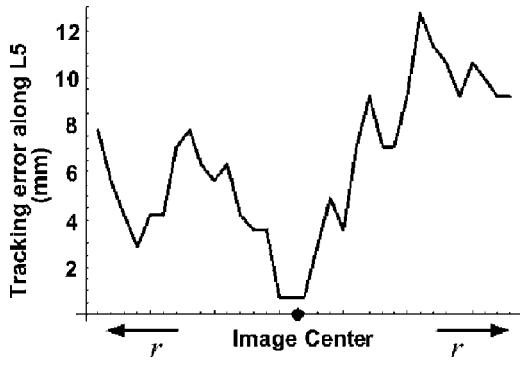

Fig. 21 Tracking errors along L5.

ments, the influence of singularity plays a more important role, so that around the central area, better performance can be achieved for tracking (see Fig. 21). Ultimately, the optimal targeting area can be modified as shown in Fig. 22.

7 Conclusion

In this paper, we have developed an enhanced 3-D tracking method using nonsingularity constraints from the interaction matrix. The influence of the system configuration on the interaction matrix has been studied, and constraints have been designed to avoid singularities of the interaction matrix. Also, we have examined the system parameters to achieve better tracking performance. Experimental results verified the effectiveness of the proposed method. Further research is underway in exploring the use of nonsingularity constraints with a robot vision system where the viewpoint can be controlled actively.

Acknowledgment

The work described in this paper was fully supported by a grant from the Research Grants Council of Hong Kong (project No. CityU117605).

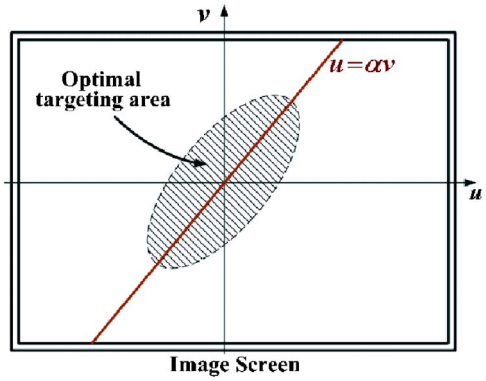

Fig. 22 The optimal targeting area.

References

1. K. Nickels and S. Hutchinson, "Model-based tracking of complex articulated objects," *IEEE Trans. Rob. Autom.* **17**(1) 28–36 (2001).
2. Y. F. Li and Z. Liu, "Information entropy based viewpoint planning for 3D object reconstruction," *IEEE Trans. Robot.* **21**(3) 324–337 (2005).
3. M. Kass, A. Witkin, and D. T. Snakes, "Active contour models," in *Proc. Int. Conf. on Computer Vision*, pp. 259–268, IEEE (1987).
4. T. Moeslund and E. Granum, "A survey of computer vision-based human motion capture," *Comput. Vis. Image Underst.* **81**, 231–268 (2001).
5. C. Collewet and F. Chaumette, "Positioning a camera with respect to planar objects of unknown shape by coupling 2-D visual servoing and 3-D estimations," *IEEE Trans. Rob. Autom.* **18**(3) 322–333 (2002).
6. N. P. Papanikolopoulos, P. K. Khosla, and T. Kanade, "Visual tracking of a moving target by a camera mounted on a robot: a combination of control and vision," *IEEE Trans. Rob. Autom.* **9**(1), 14–35 (1993).
7. Y. Mezouar, H. H. Abdelkader, P. Martinet, and F. Chaumette, "Central catadioptric visual servoing from 3D straight lines," in *Proc. 2004 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pp. 343–348 (2004).
8. B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. Rob. Autom.* **8**(6) 313–326 (1992).
9. E. Cervera, A. P. del Pobil, F. Berry, and P. Martinet, "Improving image-based visual servoing with three-dimensional features," *Int. J. Robot. Res.* **22**(10–11) 821–839 (2003).
10. F. Chaumette, "Potential problems of stability and convergence in image-based and position-based visual servoing," in *The Confluence of Vision and Control*, G. Hager and D. Kriegman, Eds., pp. 66–78, Springer-Verlag (1998).
11. H. Michel and P. Rives, "Singularities in the determination of the situation of a robot effector from the perspective view of three points," Technical Report 1850, INRIA (1993).
12. E. Malis, F. Chaumette, and S. Boudet, "2-12-D visual servoing," *IEEE Trans. Rob. Autom.* **15**(2) 238–250 (1999).
13. H. Y. Chen and Y. F. Li, "Object pose measurement by a moving camera," *in Proc. IEEE Int. Conf. on Instrumentation and Measurement Technology* (2004).

Huiying Chen received her bachelor's degree in mechatronics engineering from South China University of Technology, China. She obtained the master's degree in precision machinery engineering from the University of Tokyo, Japan. She is currently a PhD candidate in the Department of Manufacturing Engineering and Engineering Management at City University of Hong Kong. Her research interests include robot vision, visual tracking, and dynamic view planning.

Youfu Li received his BS and MS degrees in electrical engineering from Harbin Institute of Technology, China. He obtained the PhD degree from the Department of Engineering Sciences of the University of Oxford in 1993. From 1993 to 1995 he was on the research staff in the Department of Computer Sciences, University of Wales, Aberystwyth, UK. He is currently an associate professor in the Department of Manufacturing Engineering and Engineering Management at City University of Hong Kong. His research interests include robot sensing, robot vision, sensor-based control, sensor-guided manipulation, 3-D vision, visual tracking, mechatronics, and automation. He is an associate editor of IEEE Transactions on Automation Science and Engineering (T-ASE).