Available online at www.sciencedirect.com
ScienceDirect
Control Engineering Practice 15 (2007) 123–134
CONTROL ENGINEERING PRACTICE
www.elsevier.com/locate/conengprac

Adaptive extended Kalman filtering for visual motion estimation of 3D objects

Vincenzo Lippiello, Bruno Siciliano, Luigi Villani*

PRISMA Lab, Dipartimento di Informatica e Sistemistica, Università degli Studi di Napoli Federico II, Via Claudio 21, 80125 Napoli, Italy

Received 26 July 2005; accepted 23 May 2006
Available online 24 July 2006

Abstract

An algorithm for the real-time estimation of the position and orientation of a moving object of known geometry is presented in this paper. An estimation algorithm is adopted where a discrete-time extended Kalman filter computes the object pose on the basis of visual measurements of the object features. The scheme takes advantage of the prediction capability of the extended Kalman filter for the pre-selection of the features to be extracted from the image at each sample time. To enhance the robustness of the algorithm with respect to measurement noise and modelling error, an adaptive version of the extended Kalman filter, customized for visual applications, is proposed. Experimental results on a fixed single-camera visual system are presented to test the performance and the feasibility of the proposed approach.

Keywords: Pose estimation; Vision; Motion tracking; Visual servoing; Extended Kalman filter

1. Introduction

During the last decade the use of visual systems for robotics applications has become a viable option, which is often capable to offer a good cost/performance tradeoff. In fact, the effectiveness and autonomy of a robotic system operating in unstructured environments can be significantly enhanced if a visual system is adopted to achieve direct measurements of the state of the environment and of the task in progress.

Visual measurements can be directly used to perform closed-loop position control of the robot end-effector and several approaches have been proposed to the so called robot visual servoing problem (e.g., Corke, 1996; Hashimoto, 1993; Hutchinson, Hager, & Corke, 1996; Kriegman, Hager, & Morse, 1998; Vincze & Hager, 2000). Some visual servoing techniques require the computation of a tracking error defined directly in the space of the image features (Espiau, Chaumette, & Rives, 1992), or adopt hybrid solutions where some error components are defined in the image space and some others in the Cartesian space (Corke & Hutchinson, 2001; Malis, Chaumette, & Boudet, 1999). In contrast, the position-based visual servoing techniques are based on position and orientation errors defined in the Cartesian space (Thuilot, Martinet, Cordesses, & Gallice, 2002; Wilson, Hulls, & Bell, 1996), hence the (partial) 3-D reconstruction of the environment or the estimation of the pose of target objects from visual measurements is required.

The problem of reconstructing the position and orientation of moving objects in real time from image measurements has been largely investigated in the computer vision literature (Broida & Chellappa, 1986; Harris, 1992; Philip, 1991) as well as in the robotic literature (Lee & Kay, 1990; Wang & Wilson, 1992). Some of the proposed schemes are able to reconstruct in real time both the motion and the three-dimensional structure of a moving scene from its two-dimensional images (e.g., Chiuso, Favaro, Jin, & Soatto, 2002).

As a matter of fact, since visual measurements are usually affected by significant noise and disturbances due to temporal and spatial sampling and quantization of the image signal, lens distortion, etc., the accurate estimation

*Corresponding author. Tel.: +39 0817683861.
E-mail address: lvillani@unina.it (L. Villani).

0967-0661/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.conengprac.2006.05.006

of the position and orientation of an object may be a difficult task. For this reason, the extended Kalman filter (EKF) is usually adopted to achieve noise and disturbance rejection and to enhance estimation accuracy (e.g., Lee & Kay, 1990; Wang & Wilson, 1992). Kalman filtering offers many advantages over other estimation methods, e.g., temporal filtering, recursive implementation, possibility of realizing a proper statistical combination of redundant measurements, ability to change the measurement set during the operation. Also, the prediction capability of the filter allows setting up a dynamic windowing technique of the image plane which may sensibly reduce the time required for feature extraction.

If the quality of the camera sensors is good, the illumination of the scene is stable, and the model of the target object motion is accurate enough, then the use of a classic formulation of the EKF may guarantee satisfactory results. In fact, in the above scenario, it is reasonable to assume that the statistics of the state noise and of the observation noise are known a priori. On the other hand, if one or more of the above conditions are not verified, it may be convenient to adopt an adaptive extended Kalman filter (AEKF) (Myers & Tapley, 1976). In the literature various formulations of AEKF have been proposed, which address the problem of the real-time adaptation of the statistical parameters of the covariance matrices of the state and observation noise in different applications, e.g., power systems (Girgis & Peterson, 1990), mobile robots (Jetto, Longhi, & Venturini, 1999; Jetto, Longhi, & Vitali, 1999), visual pose estimation (Ficocelli & Janabi-Sharifi, 2001). The adaptive laws for the Kalman filter can be designed using different approaches, e.g., introducing additional variables to be estimated by the filter, as the noise statistics (Bai, Zhou, & Schwarz, 1998), using interacting multiple models (Bradshaw, Reid, & Murray, 1997), or adjusting the transition matrix of the Kalman filter (Wira & Urban, 2000).

The EKF is the core of a visual motion estimation algorithm presented in our previous works (Lippiello, Siciliano, & Villani, 2002; Lippiello & Villani, 2003). This algorithm can be effectively adopted for polyhedral objects, typical of industrial applications, since it is based on the use of point features (the object corners). In fact, point features are easy to identify using small windows and can be extracted with high robustness in various environmental conditions (Janabi-Sharifi & Wilson, 1997). To reduce the computational burden of the EKF and to improve estimation accuracy, the algorithm employs a computationally efficient technique for the selection of an optimal subset of feature points, among all the visible points, at each sample time. The key feature of the algorithm is the adoption of an efficient technique for representing the object geometry, based on binary space partition trees, which allows recognizing and discarding all the feature points that are occluded with respect to the camera (Drummond & Cipolla, 2002; Tarabanis, Tsai, & Kaul, 1996).

In this work, the visual motion estimation algorithm presented in Lippiello et al. (2002) and Lippiello and Villani (2003) has been enhanced by using an adaptive formulation of the EKF, to cope with uncertain or varying noise statistics. It is well known that an optimal solution for the adaptation problem of the statistical parameters of the EKF does not exist. The solution presented in this paper is based on the intuitive heuristic approach proposed in Myers and Tapley (1976), which may be formulated in a recursive manner to reduce computational time. The adaptive algorithm computes the statistics of both the state noise and the observation noise. With respect to the original work of Myers and Tapley (1976), the main contribution of this paper concerns with the adaptive law for the observation noise statistics, which has been suitably designed for a visual motion estimation problem based on the use of a variable set of image features (Lippiello & Villani, 2003). This aspect is not considered in previous works on AEKF applied to visual motion estimation (e.g., Ficocelli & Janabi-Sharifi, 2001), where only an adaptive law for the state noise covariance matrix is used.

To demonstrate the effectiveness of the proposed approach, an experimental test-bed has been developed, consisting of one fixed calibrated camera and of a robot manipulator carrying an object of known geometry. The robot allows moving the target object according to a known position and orientation trajectory, in order to measure the pose estimation error. A number of case studies are presented, to compare the performance of the EKF versus the AEKF using different object trajectories as well as to appreciate the effects of the update laws for the state and observation noise covariance matrices separately.

The paper is organized as follows. In Section 2, the pin-hole model for a fixed camera and some relevant reference frames are introduced. The basic equations of the EKF are briefly reviewed in Section 3, while the equations of the AEKF are presented in Section 4. The visual motion estimation algorithm adopted in the paper is briefly presented in Section 5. Finally, Section 6 presents the experimental case studies and Section 7 reports the conclusions. The equations used in the Kalman filter can be found in the Appendix.

2. Modeling

Consider the pin-hole model of a video camera fixed with respect to a base coordinate frame O-xyz represented in Fig. 1. Let O_c-$x_c y_c z_c$ be a frame attached to the camera (camera frame), with the z_c-axis aligned to the optical axis and the origin in the optical center. In the following, a superscript will be used to denote the reference frame of a variable, when different from the base frame.

The sensor plane is parallel to the $x_c y_c$-plane at a distance $-f_e$ along the z_c-axis, where f_e is the effective focal length of the camera lens. The image plane is parallel to the $x_c y_c$-plane at a distance f_e along the z_c-axis. The intersection of the optical axis with the image plane defines

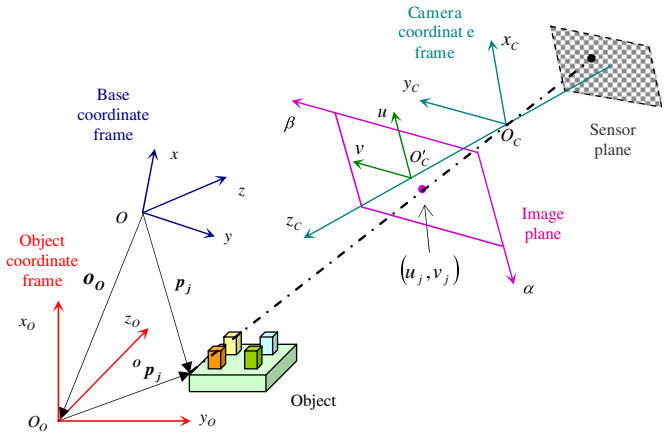

Fig. 1. Pin-hole model of the camera and reference frames.

the principal optic point O'_c, which is the origin of the image frame O'_c-uv whose axes u and v are taken parallel to the axes x_c and y_c, respectively.

A point P with coordinates ${}^c\boldsymbol{p} = [{}^cx\ {}^cy\ {}^cz]^{\mathrm{T}}$ in the camera frame is projected onto the point of the image plane whose coordinates can be computed with the equation

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{f_e}{{}^cz} \begin{bmatrix} {}^cx \\ {}^cy \end{bmatrix} \tag{1}$$

known as perspective transformation.

Consider an object frame O_0-$x_0y_0z_0$ attached to the target object. The position and orientation of the object frame with respect to the base frame can be expressed in terms of the coordinate vector of the origin $\boldsymbol{o}_0 = [x_0\ y_0\ z_0]^{\mathrm{T}}$ and of the rotation matrix $\boldsymbol{R}_0(\boldsymbol{\phi}_0)$, where $\boldsymbol{\phi}_0 = [\varphi_0\ \vartheta_0\ \psi_0]^{\mathrm{T}}$ is the vector of the Roll, Pitch and Yaw angles.

Consider m feature points of the object. The coordinate vector $\boldsymbol{p}_j$ of the feature point P_j $(j = 1, \ldots, m)$ can be expressed in the base frame as

$$\boldsymbol{p}_j = \boldsymbol{o}_0 + \boldsymbol{R}_0(\boldsymbol{\phi}_0)\,{}^o\boldsymbol{p}_j, \tag{2}$$

where ${}^o\boldsymbol{p}_j$ is the coordinate vector of P_j expressed in the object frame. Note that ${}^o\boldsymbol{p}_j$ is a constant vector that is assumed to be known, since it can be computed from a CAD model of the object or via a suitable calibration procedure. The coordinate vector ${}^c\boldsymbol{p}_j$ of P_j with respect to the camera frame can be computed as

$${}^c\boldsymbol{p}_j = \boldsymbol{R}_c^{\mathrm{T}}(\boldsymbol{p}_j - \boldsymbol{o}_c), \tag{3}$$

where $\boldsymbol{o}_c$ and $\boldsymbol{R}_c$ are, respectively, the position vector and the rotation matrix of the camera frame with respect to the base frame. These quantities are constant, because the camera is assumed to be fixed to the workspace, and can be computed through a suitable calibration procedure (Weng, Cohen, & Herniou, 1992).

By folding the $3m$ equations (2) and (3) into the perspective transformation (1), a system of $2m$ nonlinear equations is achieved. The equations depend on the measurements of the m feature points in the image plane of the camera, while the six components of the vectors $\boldsymbol{o}_0$ and $\boldsymbol{\phi}_0$ are the unknown quantities to be estimated. To achieve a unique solution at least four non-aligned points are required.

The computation of the solution is non-trivial and for visual motion estimation and it has to be repeated at a high sampling rate. The Kalman filter provides a computationally tractable solution, which can also incorporate and exploit redundant measurement information.

3. Extended Kalman filter

In order to estimate the pose of the object, a discrete-time state space model of the object motion has to be considered. The state vector of the model is chosen as the (12×1) vector

$$\boldsymbol{w} = [\,x_0\ \dot{x}_0\ y_0\ \dot{y}_0\ z_0\ \dot{z}_0 \quad \varphi_0\dot{\varphi}_0\vartheta_0\dot{\vartheta}_0\psi_0\dot{\psi}_0\,]^{\mathrm{T}}. \tag{4}$$

For simplicity, the object velocity is assumed to be constant over one sample time interval T. This approximation is reasonable in the hypothesis that T is sufficiently small. The corresponding dynamic modelling error can be considered as an input disturbance $\boldsymbol{\gamma}_k$. The discrete-time dynamic model can be written as

$$\boldsymbol{w}_k = \boldsymbol{A}\boldsymbol{w}_{k-1} + \boldsymbol{\gamma}_k, \tag{5}$$

where the state transition matrix $\boldsymbol{A}$ is a constant (12×12) block diagonal matrix of the form

$$\boldsymbol{A} = \mathrm{diag}\left\{ \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \ldots, \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \right\},$$

which is achieved using the well-known Euler method, i.e., computing the positions from velocities on the basis of a first-order approximation to the Taylor series.

The outputs of the Kalman filter are chosen as the vectors of the normalized coordinates of the m feature points in the image plane of the camera

$$\boldsymbol{\zeta}_k^u = \left[\frac{u_1}{f_e} \quad \cdots \quad \frac{u_m}{f_e} \right]_k^{\mathrm{T}}, \tag{6a}$$

$$\boldsymbol{\zeta}_k^v = \left[\frac{v_1}{f_e} \quad \cdots \quad \frac{v_m}{f_e} \right]_k^{\mathrm{T}}. \tag{6b}$$

In view of (1), the corresponding output model can be written in the form

$$\boldsymbol{\zeta}_k^u = \boldsymbol{g}^u(\boldsymbol{w}_k) + \boldsymbol{v}_k^u, \tag{7a}$$

$$\boldsymbol{\zeta}_k^v = \boldsymbol{g}^v(\boldsymbol{w}_k) + \boldsymbol{v}_k^v, \tag{7b}$$

where $\boldsymbol{v}_k^u$ and $\boldsymbol{v}_k^v$ are the observation noise vectors for the u and v components of the normalized image plane, whereas

the vector functions $\boldsymbol{g}^u(\boldsymbol{w}_k)$ and $\boldsymbol{g}^v(\boldsymbol{w}_k)$ are defined as

$$\boldsymbol{g}^u(\boldsymbol{w}_k) = \begin{bmatrix} \frac{^c x_1}{^c z_1} & \cdots & \frac{^c x_m}{^c z_m} \end{bmatrix}_k^{\mathrm{T}}, \tag{8a}$$

$$\boldsymbol{g}^v(\boldsymbol{w}_k) = \begin{bmatrix} \frac{^c y_1}{^c z_1} & \cdots & \frac{^c y_m}{^c z_m} \end{bmatrix}_k^{\mathrm{T}}. \tag{8b}$$

The coordinates of the feature points $^c\boldsymbol{p}_j$ in Eqs. (8) are computed from the state vector $\boldsymbol{w}_k$ via Eqs. (2), (3).

The components of the disturbance quantities $\boldsymbol{\gamma}_k$, $\boldsymbol{v}_k^u$ and $\boldsymbol{v}_k^v$ are considered as independent, non-stationary, Gaussian, white noise sequences with the statistical properties

$$E[\boldsymbol{\gamma}_k] = \boldsymbol{q}_k, \tag{9a}$$

$$E[\boldsymbol{v}_k^u] = \boldsymbol{r}_k^u, \tag{9b}$$

$$E[\boldsymbol{v}_k^v] = \boldsymbol{r}_k^v, \tag{9c}$$

$$E[(\boldsymbol{\gamma}_k - \boldsymbol{q}_k)(\boldsymbol{\gamma}_l - \boldsymbol{q}_l)^{\mathrm{T}}] = \boldsymbol{Q}_k \delta_{kl}, \tag{9d}$$

$$E[(\boldsymbol{v}_k^u - \boldsymbol{r}_k^u)(\boldsymbol{v}_l^u - \boldsymbol{r}_l^u)] = \boldsymbol{R}_k^u \delta_{kl}, \tag{9e}$$

$$E[(\boldsymbol{v}_k^v - \boldsymbol{r}_k^v)(\boldsymbol{v}_l^v - \boldsymbol{r}_l^v)] = \boldsymbol{R}_k^v \delta_{kl}, \tag{9f}$$

where $E[\cdot]$ indicates the statistical mean operator applied to the components of a vector or matrix, and δ is the Kroneker symbol.

Since the output model is nonlinear in the system state, the output equations must be linearized about the current state estimate at each sample time. This leads to the so-called EKF.

The following recursive algorithm is the best linear, minimum variance, unbiased estimator of the state vector for the system defined by Eqs. (5), (7).

The first step of the algorithm provides an optimal estimate of the state at the next sample time according to the recursive equations

$$\boldsymbol{w}_{k,k-1} = \boldsymbol{A}\boldsymbol{w}_{k-1,k-1} + \boldsymbol{q}_{k-1}, \tag{10a}$$

$$\boldsymbol{P}_{k,k-1} = \boldsymbol{A}\boldsymbol{P}_{k-1,k-1}\boldsymbol{A}^{\mathrm{T}} + \boldsymbol{Q}_{k-1}, \tag{10b}$$

where $\boldsymbol{w}_{k,k-1}$ is the propagated state vector and $\boldsymbol{P}_{k,k-1}$ is the (12×12) covariance matrix conditioned on observations prior to time k. The second step improves the previous estimate by using the input measurements according to the equations

$$\boldsymbol{w}_{k,k} = \boldsymbol{w}_{k,k-1} + [\boldsymbol{K}_k^u \;\; \boldsymbol{K}_k^v] \begin{bmatrix} \boldsymbol{\zeta}_k^u - \boldsymbol{g}^u(\boldsymbol{w}_{k,k-1}) - \boldsymbol{r}_k^u \\ \boldsymbol{\zeta}_k^v - \boldsymbol{g}^v(\boldsymbol{w}_{k,k-1}) - \boldsymbol{r}_k^v \end{bmatrix}, \tag{11a}$$

$$\boldsymbol{P}_{k,k} = \boldsymbol{P}_{k,k-1} - [\boldsymbol{K}_k^u \;\; \boldsymbol{K}_k^v] \begin{bmatrix} \boldsymbol{H}_k^u \\ \boldsymbol{H}_k^v \end{bmatrix} \boldsymbol{P}_{k,k-1}, \tag{11b}$$

where $\boldsymbol{K}_k^u$ and $\boldsymbol{K}_k^v$ are the $(12 \times m)$ Kalman matrix gains

$$\boldsymbol{K}_k^u = \boldsymbol{P}_{k,k-1}\boldsymbol{H}_k^{u\mathrm{T}}(\boldsymbol{R}_k^u + \boldsymbol{\Gamma}_k^u)^{-1}, \tag{12a}$$

$$\boldsymbol{K}_k^v = \boldsymbol{P}_{k,k-1}\boldsymbol{H}_k^{v\mathrm{T}}(\boldsymbol{R}_k^v + \boldsymbol{\Gamma}_k^v)^{-1}, \tag{12b}$$

being $\boldsymbol{H}_k^u$ and $\boldsymbol{H}_k^v$ the $(m \times 12)$ Jacobian matrices of the output vector functions

$$\boldsymbol{H}_k^u = \left.\frac{\partial \boldsymbol{g}^u(\boldsymbol{w})}{\partial \boldsymbol{w}}\right|_{\boldsymbol{w}=\boldsymbol{w}_{k,k-1}}, \tag{13a}$$

$$\boldsymbol{H}_k^v = \left.\frac{\partial \boldsymbol{g}^v(\boldsymbol{w})}{\partial \boldsymbol{w}}\right|_{\boldsymbol{w}=\boldsymbol{w}_{k,k-1}}, \tag{13b}$$

and $\boldsymbol{\Gamma}_k^u$ and $\boldsymbol{\Gamma}_k^v$ defined as

$$\boldsymbol{\Gamma}_k^u = \boldsymbol{H}_k^u \boldsymbol{P}_{k,k-1}\boldsymbol{H}_k^{u\mathrm{T}}, \tag{14a}$$

$$\boldsymbol{\Gamma}_k^v = \boldsymbol{H}_k^v \boldsymbol{P}_{k,k-1}\boldsymbol{H}_k^{v\mathrm{T}}. \tag{14b}$$

Eqs. (13) correspond to the linearization of the output Kalman model about the last predicted value of the state $\boldsymbol{w}_{k,k-1}$, as required in the formulation of the EKF (Lewis, 1986).

Note that a prior estimate of the state $\boldsymbol{w}_0$ and of the state covariance $\boldsymbol{P}_0$ and a prior statistical information represented by Eqs. (9) are required. The analytic expressions of $\boldsymbol{H}_k^u$ and $\boldsymbol{H}_k^v$ can be found in the Appendix.

It is worth remarking that the estimation error obviously depends on the accuracy of the model of the object motion. The main factors affecting performance are: the order of the model (first order vs. second order), the selection of the statistical parameters of the EKF, the number and the spatial distribution of the feature points of the object on the image plane (depending on the object dimension and on its distance from the camera) as well as the value of the sample time T compared to the object velocity.

In this paper, a reduced order model of the object motion is adopted, in order to decrease the computational complexity of the EKF equations. Moreover, the statistical parameters of the filter are updated according to a suitable adaptive algorithm. Finally, a dynamic selection strategy is adopted to compute, at each sample time, an optimal and minimal subset of feature points. This allows the sample time to be set as the minimum value compatible with the hardware constraint imposed by the camera frame rate.

4. Adaptive extended Kalman filter

If a high-quality camera sensor is used, the illumination of the scene is stable, and the velocity of the tracked object is nearly constant over a sample time T, then it is possible to use constant statistical parameters with optimal results. On the other hand, if these conditions are not satisfied, it may be convenient to update in real time the statistical parameters $\{\boldsymbol{q}_k, \boldsymbol{Q}_k, \boldsymbol{r}_k^u, \boldsymbol{r}_k^v, \boldsymbol{R}_k^u, \boldsymbol{R}_k^v\}$. This leads to the AEKF.

It is known that an optimal estimator of the statistical parameters cited above does not exist, but many sub-optimal schemes have been proposed. In this work the intuitive approach proposed in Myers and Tapley (1976) and revised in Girgis and Peterson (1990), Jetto, Longhi, and Venturini (1999), Jetto, Longhi, and Vitali (1999) and Ficocelli and Janabi-Sharifi (2001) is redefined for a visual

motion estimation problem. The adaptive algorithm is formulated in a recursive, limited memory format.

The basic hypothesis for this approach is the constant value of the statistical parameters over N sample times (Myers & Tapley, 1976).

Since not all the visual features are always available during the motion and their location into the scene is strongly variable, it may be reasonable to assume the statistics of the observation noise to be equal for all the measurements of the feature points in the scene at time k. This assumption can be removed by computing the statics for each visual feature separately, even though better results may be not guaranteed. Hence the quantities $\{\boldsymbol{r}_k^u, \boldsymbol{r}_k^v, \boldsymbol{R}_k^u, \boldsymbol{R}_k^v\}$ are replaced by the quantities $\{r_k^u \boldsymbol{\iota}_m, r_k^v \boldsymbol{\iota}_m, \sigma_k^{u2} \boldsymbol{I}_m, \sigma_k^{v2} \boldsymbol{I}_m\}$, where $\boldsymbol{\iota}_m$ indicates a $(m \times 1)$ vector of components equal to 1 and $\boldsymbol{I}_m$ indicates the $(m \times m)$ identity matrix. Moreover, the samples of the observation noise sequences $\boldsymbol{v}_i^u$ $(\boldsymbol{v}_i^v)$ are independent for $i = 1, \ldots, N$ and have a Gaussian distribution with mean $r^u \boldsymbol{\iota}_m$ $(r^v \boldsymbol{\iota}_m)$ and variance $\sigma^{u2} \boldsymbol{I}_m$ $(\sigma^{v2} \boldsymbol{I}_m)$, where the parameters r^u, r^v, σ^u and σ^v are constant over N sample times.

In view of the nonlinear relation (7), an intuitive approximation of the observation noise sample vectors at time k is given by the quantities

$$\boldsymbol{\rho}_k^u = \boldsymbol{\zeta}_k^u - \boldsymbol{g}^u(\boldsymbol{w}_{k,k-1}), \tag{15a}$$

$$\boldsymbol{\rho}_k^v = \boldsymbol{\zeta}_k^v - \boldsymbol{g}^v(\boldsymbol{w}_{k,k-1}), \tag{15b}$$

which can be considered as independent and identically distributed over N samples. It can be shown (Myers & Tapley, 1976) that an unbiased estimator for r^u and r^v can be taken as

$$\hat{r}^u = \frac{1}{N} \sum_{i=1}^{N} \overline{\rho}_i^u, \tag{16a}$$

$$\hat{r}^v = \frac{1}{N} \sum_{i=1}^{N} \overline{\rho}_i^v, \tag{16b}$$

where $\overline{\rho}_i^u$ and $\overline{\rho}_i^v$ are scalar quantities equal to the mean values of the components of the vectors $\boldsymbol{\rho}_i^u$ and $\boldsymbol{\rho}_i^v$, respectively. Moreover, an unbiased estimator for σ^{u2} and σ^{v2} may be obtained as

$$\hat{\sigma}^{u2} = \frac{1}{m(N-1)} \sum_{i=1}^{N} \left\{ \|\boldsymbol{\rho}_i^u - \hat{r}^u \boldsymbol{\iota}_m\|^2 - \frac{N-1}{N} \operatorname{tr}(\boldsymbol{\Gamma}_i^u) \right\}, \tag{17a}$$

$$\hat{\sigma}^{v2} = \frac{1}{m(N-1)} \sum_{i=1}^{N} \left\{ \|\boldsymbol{\rho}_i^v - \hat{r}^v \boldsymbol{\iota}_m\|^2 - \frac{N-1}{N} \operatorname{tr}(\boldsymbol{\Gamma}_i^v) \right\}. \tag{17b}$$

where the function $\operatorname{tr}(\cdot)$ is the trace of the input matrix.

For the state noise statistics, in view of the linear dynamic state relation at time k given by (5), an intuitive approximation for the state noise vector at time k is

$$\boldsymbol{\varrho}_k = \boldsymbol{w}_k - \boldsymbol{A} \boldsymbol{w}_{k,k-1}, \tag{18}$$

which may be considered independent and identically distributed over N samples. As before, it can be shown that an unbiased estimator for the mean value $\boldsymbol{q}$ of the state noise may be obtained as

$$\hat{\boldsymbol{q}} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{\varrho}_i, \tag{19}$$

while an unbiased estimator for the covariance matrix $\boldsymbol{Q}$ is given by

$$\hat{\boldsymbol{Q}} = \frac{1}{N-1} \sum_{i=1}^{N} \left\{ (\boldsymbol{\varrho}_i - \hat{\boldsymbol{q}})(\boldsymbol{\varrho}_i - \hat{\boldsymbol{q}})^{\mathrm{T}} - \frac{N-1}{N} \boldsymbol{\Delta}_i \right\}, \tag{20}$$

where $\boldsymbol{\Delta}_i = \boldsymbol{A} \boldsymbol{P}_{i,i-1} \boldsymbol{A}^{\mathrm{T}} - \boldsymbol{P}_{i,i}$.

In sum, Eqs. (15)–(20) provide a heuristic unbiased estimator for the statistical parameters of a EKF used for visual motion estimation, on the assumption that the last N samples are statistically independent and identically distributed.

5. Recursive AEKF

Using the previous results, a recursive limited memory formulation of the AEKF may be designed. The required prior knowledge information is represented by an initial estimate of the quantities $\boldsymbol{w}_{1,0}$, $\boldsymbol{P}_{1,0}$, $\hat{\boldsymbol{q}}_0$, $\hat{\boldsymbol{Q}}_0$, $\hat{r}_0^u$, $\hat{r}_0^v$, $\hat{\sigma}_0^u$ and $\hat{\sigma}_0^v$.

For generality, it is assumed that the observation noise statistical parameters are constant over N_r time samples while the state noise statistical parameters are constant over N_q time samples.

The first step of the recursive algorithm is the linearization of the output Kalman model about the last predicted value of the state $\boldsymbol{w}_{k,k-1}$, according to (13), and the computation of the matrices $\boldsymbol{\Gamma}_k^u$ and $\boldsymbol{\Gamma}_k^v$ in (14).

Starting from the time N_r, the second step is the evaluation of the current noise vector and the computation of the estimated observation noise statistics as follows:

$$\boldsymbol{\rho}_k^u = \boldsymbol{\zeta}_k^u - \boldsymbol{g}^u(\boldsymbol{w}_{k,k-1}), \tag{21a}$$

$$\boldsymbol{\rho}_k^v = \boldsymbol{\zeta}_k^v - \boldsymbol{g}^v(\boldsymbol{w}_{k,k-1}), \tag{21b}$$

$$\hat{r}_k^u = \hat{r}_{k-1}^u + \frac{1}{N_r} (\overline{\rho}_k^u - \overline{\rho}_{k-N_r}^u), \tag{21c}$$

$$\hat{r}_k^v = \hat{r}_{k-1}^v + \frac{1}{N_r} (\overline{\rho}_k^v - \overline{\rho}_{k-N_r}^v), \tag{21d}$$

$$\hat{\sigma}_k^{u2} = \hat{\sigma}_{k-1^2}^u + \frac{1}{m(N_r-1)} \left\{ \|\boldsymbol{\rho}_k^u - \hat{r}_k^u \boldsymbol{\iota}_m\|^2 - \|\boldsymbol{\rho}_{k-N_r}^u - \hat{r}_k^u \boldsymbol{\iota}_m\|^2 + \frac{1}{N_r} \|\boldsymbol{\rho}_k^u - \boldsymbol{\rho}_{k-N_r}^u\|^2 + \frac{N_r - 1}{N_r} \operatorname{tr}(\boldsymbol{\Gamma}_{k-N_r}^u - \boldsymbol{\Gamma}_k^u) \right\}, \tag{21e}$$

$$\hat{\sigma}_k^{v2} = \hat{\sigma}_{k-1^2}^v + \frac{1}{m(N_r-1)} \left\{ \|\boldsymbol{\rho}_k^v - \hat{r}_k^v \boldsymbol{\iota}_m\|^2 - \|\boldsymbol{\rho}_{k-N_r}^v - \hat{r}_k^v \boldsymbol{\iota}_m\|^2 + \frac{1}{N_r} \|\boldsymbol{\rho}_k^v - \boldsymbol{\rho}_{k-N_r}^v\|^2 + \frac{N_r - 1}{N_r} \operatorname{tr}(\boldsymbol{\Gamma}_{k-N_r}^v - \boldsymbol{\Gamma}_k^v) \right\}. \tag{21f}$$

The third step consists in the evaluation of the Kalman gains

$$\boldsymbol{K}_k^u = \boldsymbol{P}_{k,k-1} \boldsymbol{H}_k^{u\mathrm{T}} (\boldsymbol{\Gamma}_k^u + \hat{\sigma}_k^{u2} \boldsymbol{I}_m)^{-1}, \tag{22a}$$

$$\boldsymbol{K}_k^v = \boldsymbol{P}_{k,k-1} \boldsymbol{H}_k^{v\mathrm{T}} (\boldsymbol{\Gamma}_k^v + \hat{\sigma}_k^{v2} \boldsymbol{I}_m)^{-1}, \tag{22b}$$

while the fourth step is the state correction on the basis of the current measurements

$$\boldsymbol{w}_{k,k} = \boldsymbol{w}_{k,k-1} + [\boldsymbol{K}_k^u \;\; \boldsymbol{K}_k^v] \begin{bmatrix} \boldsymbol{\rho}_k^u - \hat{r}_k^u \boldsymbol{\iota}_m \\ \boldsymbol{\rho}_k^v - \hat{r}_k^v \boldsymbol{\iota}_m \end{bmatrix}, \tag{23a}$$

$$\boldsymbol{P}_{k,k} = \boldsymbol{P}_{k,k-1} - [\boldsymbol{K}_k^u \;\; \boldsymbol{K}_k^v] \begin{bmatrix} \boldsymbol{H}_k^u \\ \boldsymbol{H}_k^v \end{bmatrix} \boldsymbol{P}_{k,k-1}. \tag{23b}$$

Starting from the time N_q, the fifth step is the evaluation of the current state noise vector and the computation of the estimated state noise statistics as follows:

$$\boldsymbol{\varrho}_k = \boldsymbol{w}_{k,k} - \boldsymbol{A}\boldsymbol{w}_{k,k-1}, \tag{24a}$$

$$\boldsymbol{\Delta}_k = \boldsymbol{A}\boldsymbol{P}_{k,k-1}\boldsymbol{A}^{\mathrm{T}} - \boldsymbol{P}_{k,k}, \tag{24b}$$

$$\hat{\boldsymbol{q}}_k = \hat{\boldsymbol{q}}_{k-1} + \frac{1}{N_q}(\boldsymbol{\varrho}_k - \boldsymbol{\varrho}_{k-N_q}), \tag{24c}$$

$$\begin{aligned} \hat{\boldsymbol{Q}}_k = \hat{\boldsymbol{Q}}_{k-1} + \frac{1}{N_q - 1} \Big\{ & (\boldsymbol{\varrho}_k - \hat{\boldsymbol{q}}_k)(\boldsymbol{\varrho}_k - \hat{\boldsymbol{q}}_k)^{\mathrm{T}} \\ & - (\boldsymbol{\varrho}_{k-N_q} - \hat{\boldsymbol{q}}_k)(\boldsymbol{\varrho}_{k-N_q} - \hat{\boldsymbol{q}}_k)^{\mathrm{T}} \\ & + \frac{1}{N_q}(\boldsymbol{\varrho}_k - \boldsymbol{\varrho}_{k-N_q})(\boldsymbol{\varrho}_k - \boldsymbol{\varrho}_{k-N_q})^{\mathrm{T}} \\ & + \frac{N_q - 1}{N_q}(\boldsymbol{\Delta}_{k-N_q} - \boldsymbol{\Delta}_k) \Big\}. \end{aligned} \tag{24d}$$

The sixth and last step consists in the evaluation of the predicted state for the next sample time

$$\boldsymbol{w}_{k+1,k} = \boldsymbol{A}\boldsymbol{w}_{k,k} + \hat{\boldsymbol{q}}_k, \tag{25a}$$

$$\boldsymbol{P}_{k+1,k} = \boldsymbol{A}\boldsymbol{P}_{k,k}\boldsymbol{A}^{\mathrm{T}} + \hat{\boldsymbol{Q}}_k. \tag{25b}$$

Note that the update of the noise statistics starts from time N_r for the observation noise and from time N_q for the state noise. Before such times, those quantities are constant and equal to the initial values.

6. Visual motion estimation algorithm

The accuracy of the estimate provided by the Kalman filter depends on the number of the available feature points. Inclusion of extra points may improve the estimation accuracy but increases the computational cost. It has been shown that a number of feature points between four and six, if properly chosen, may represent a good trade-off (Wang & Wilson, 1992). Automatic selection algorithms have been developed to find the optimal feature points (Feddema, Lee, & Mitchell, 1991; Janabi-Sharifi & Wilson, 1997). In the following, an efficient selection technique proposed in Lippiello et al. (2002) is adopted, which exploits the prediction of the object pose provided by the Kalman filter to perform a pre-selection of the points that are visible to the camera at the next sample time. A detailed presentation of the algorithm, which is based on the binary space partitioning (BSP) tree structure to represent the object geometry, can be found in Lippiello et al. (2002) and Lippiello and Villani (2003).

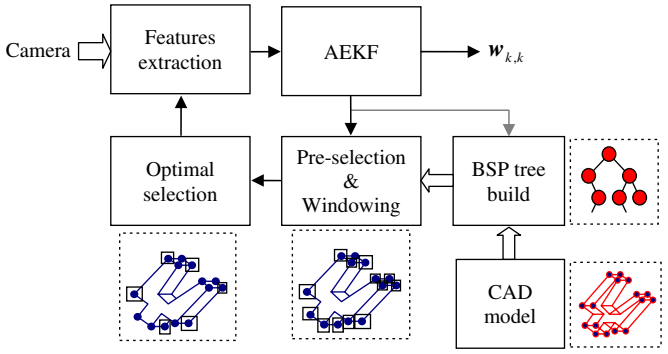

Fig. 2. Block scheme of the visual motion estimation algorithm.

In this paper, only a short description of the algorithm, which is sketched in the closed loop estimation scheme of Fig. 2, is provided. It can be seen that the prediction of object pose at next sample time ($\boldsymbol{w}_{k+1,k}$) provided by the Kalman filter is fed back to the pre-selection algorithm which, on the basis of a BSP representation of the object, is capable to find and locate all the feature points which are expected to be visible from the camera. Then, an optimal selection algorithm is used to choose an optimal subset of feature points and compute the size and location of the corresponding windows on the image plane to be grabbed for image processing. Finally, the feature extraction algorithm provides the effective position of the feature points on the image plane of the camera to be input to the Kalman filter which computes the actual object pose $\boldsymbol{w}_{k,k}$.

The above algorithm allows the nice features of Kalman filtering to be fully exploited. In fact, the AEKF provides robustness with respect to visual measurement noise and changing in the lighting conditions. Moreover, the closed loop structure allows achieving robustness with respect to the loss of feature points caused by occlusion or exit from the visible space of the camera. Finally, the use of point-type features combined with the optimal selection and windowing algorithms allows minimizing the computational time required for image processing.

7. Experiments

The effectiveness and the performance of the proposed visual motion estimation algorithm have been experimentally tested using a single-camera visual system.

The experimental set-up is composed by a PC with Pentium IV 1.7 GHz processor equipped with a MATROX Genesis board, a SONY 8500CE B/W camera, and a COMAU Smart3-S robot. The MATROX board is used as frame grabber and for a partial image processing (e.g., windows extraction from the image). The PC host is also used to realize the whole BSP structure management, the pre-selection algorithm, the selection algorithm, the dynamic windowing and the Kalman filtering. Some steps of image processing have been parallelized on the MATROX board and on the PC, so as to reduce computational time.

The robot is used to move an object in the visual space of the camera; thus the object position and orientation with respect to the base frame of the robot can be computed from joint position measurements via the direct kinematic equation. In order to test the accuracy of the estimation provided by the Kalman filter, the camera was calibrated with respect to the base frame of the robot using a suitable calibration procedure presented in Weng et al. (1992), where the robot is used to place a calibration pattern in some known pose of the visible space of the camera.

The camera resolution is 576×763 pixels and the nominal focal length of the lenses is 16 mm. The camera is disposed at a distance of about 130 cm from the object. The sample time used for estimation is $T = 0.04$ s, which is the minimum time allowed by the camera frame rate (about 26 fps). This time is more than enough for feature extraction and pose estimation. A simple neon illumination has been used, in order to test the robustness of the setup in the case of noisy visual measurements. In fact, during the object motion, the local illumination conditions of the windows of the image plane selected for feature extraction are quite variable due to reflections or shadows.

The image features are the corners of the object, which can be extracted with high robustness in various environmental conditions. The feature extraction algorithm is based on Canny's method for edge detection (Canny, 1986) and on a simple custom implementation of a corner detector. The object used in the experiment has 40 corners, which are all candidate for feature extraction. Fig. 3 shows the stereo vision system and the robot carrying the object. Note that only the left camera is used in the experiments.

For both EKF and AEKF, the initial value of the matrix $\boldsymbol{P}_{1,0}$ has been chosen as the null matrix; moreover, the initial value of the state vector $\boldsymbol{w}_{1,0}$ has been set null for the velocity components, while the pose components have been roughly estimated through direct measurements.

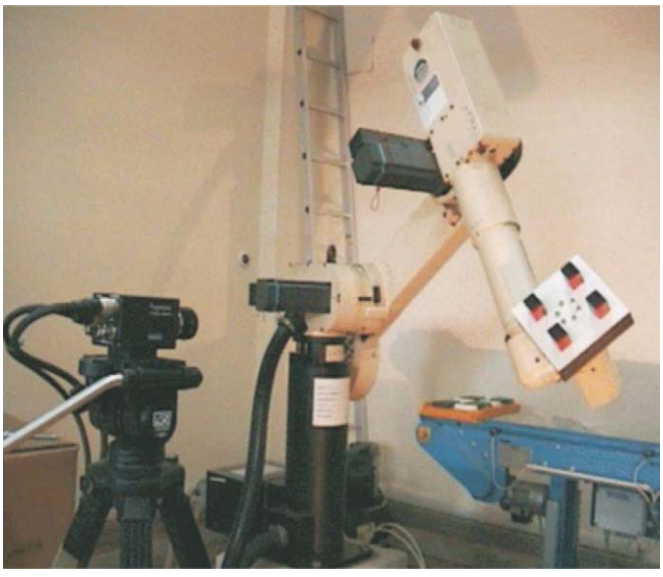

Fig. 3. COMAU Smart3-S robot carrying the moving object and SONY 8500CE camera.

The covariance matrix $\boldsymbol{Q}$ has been chosen as a diagonal matrix, both in the non-adaptive and in the adaptive case; moreover, to prevent some typical implementation problems of Kalman filters, some of the modifications used in Myers and Tapley (1976) have been adopted.

The values of the statistical parameters used for the EKF are set as initial values for the AEKF; they are:

$$\hat{r}_0^u = \hat{r}_0^v = 0,$$
$$\hat{\sigma}_0^{u2} = \hat{\sigma}_0^{v2} = 9.0,$$
$$\hat{\boldsymbol{q}}_0 = \boldsymbol{0},$$
$$\hat{\boldsymbol{Q}}_0 = \mathrm{diag}\{0,5,0,5,0,5,0,20,0,20,0,20\} \cdot 10^{-6}.$$

The physical dimensions are: pixel and pixel^2, respectively, for the mean and variances of the observation noise; mm, mm/s, rad and rad/s for the components of the mean of the state noise; mm^2, (mm/s)^2, rad^2 and (rad/s)^2 for the corresponding covariances.

The initial values of the observation noise covariances have been evaluated during the camera calibration procedure while the initial values of the state noise covariances have been set on the basis of the acceleration range of the object trajectories. In fact, in view of the hypothesis that the velocity is constant over a sample time T, modelling errors are mainly due to the change of velocity. These values have been further tuned on the basis of a set of experiments carried out using the EKF, to achieve satisfactory tracking performance.

Note that all the elements of the covariance matrix $\hat{\boldsymbol{Q}}_0$ corresponding to the position components of the state have been considered initially zero for the AEKF and constantly zero for the EKF. Moreover, the values $N_q = N_r = 30$ have been chosen by trial and error as a result of a trade-off between significance of statistical information (requiring the maximum number of samples) and adaptation capabilities (requiring the minimum number of samples).

Two different case studies are considered. The first is aimed at comparing the performance of the EKF to that of the AEKF considering different object trajectories. The second is aimed at evaluating the effects of the update laws for the matrix $\boldsymbol{Q}$ and $\boldsymbol{R}$ separately, using the same object trajectory. It should be remarked that the adoption of the AEKF in lieu of the EKF causes only a modest increase of the computational cost that, in terms of overall processor time, is about 16%. This result is in accordance to those reported in Myers and Tapley (1976) and Ficocelli and Janabi-Sharifi (2001).

7.1. First case study: EKF vs. AEKF

Three different object trajectories are considered, both for the EKF and the AEKF:

- *TrajP*: The object position varies according to the time history reported in Fig. 4(a) and (c), while the object

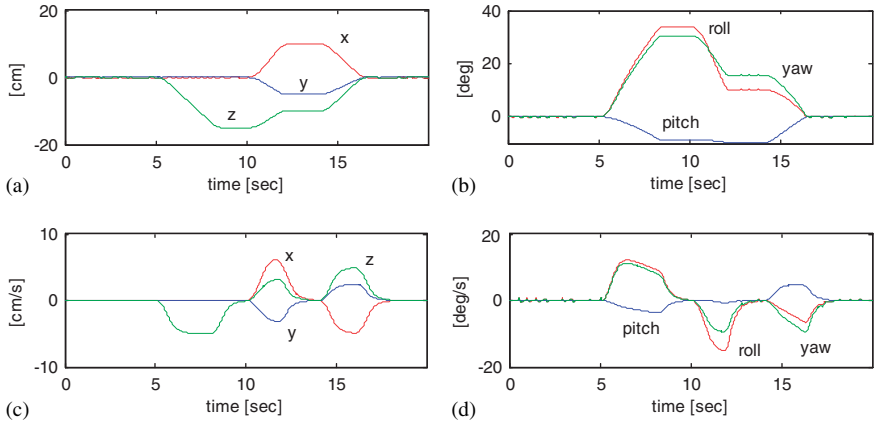

Fig. 4. Time history of the object trajectory *TrajPO*: (a) position vector; (b) RPY angles; (c) linear velocity; (d) time derivative of RPY angles.

orientation is left constant. The norm of the maximum linear velocity is about 10 cm/s.

- *TrajO*: The object position is left constant while the object orientation varies according to the time history reported in Fig. 4(b) and (d). The norm of the maximum velocity for the RPY angles is about 20 deg/s.
- *TrajPO*: The object position and orientation varies according to the time history of Fig. 4.

Note the trajectory *TrajPO* is the composition of *TrajP* and *TrajO*, but the resulting trajectories of the feature points on the image plane are different in the three cases. Hence the considered trajectories represent a significant test base to make a comparison.

The results of the experiments are summarized in Table 1 where the mean value and the standard deviation of the norm of the position error $\boldsymbol{e}_P$ and orientation error $\boldsymbol{e}_O$ are reported, together with the relative improvement achieved using the AEKF with respect to the EKF. The pose estimation error is defined as the difference between the "true" object pose computed from the robot forward kinematics and the object pose estimated by the Kalman filter. It can be seen that the use of the AEKF allows a general improvement of the tracking performance, especially for the mean value and the standard deviation of the orientation error components.

For the trajectory *TrajPO*, the time history of the components of the pose estimation errors in the base frame are shown in Fig. 5. It can be seen that the initial values of the errors are the same in the case of the EKF and the AEKF. In particular, there is an initial offset for the x-component of the position error, due to the initial misalignment between the real initial position and the initial state of the Kalman filter. This error is recovered by the Kalman filter during the first 5 s, in the absence of motion. During the motion, the position errors keep limited for all the components, but is higher for the x-component. In fact, in the experimental set up, the x-axis is aligned to the optical axis of the camera, thus the evaluation of the x-component of the object position, for a single-camera system, is more sensitive to measurements

Table 1
Comparison of the pose errors for EKF and AEKF in the first case study

		EKF		AEKF		Improvement	
		Mean	StD	Mean	StD	Mean (%)	StD (%)
TrajP	$\boldsymbol{e}_P$ (mm)	13.71	11.16	12.59	10.89	8.2	2.4
	$\boldsymbol{e}_O$ (deg)	3.50	3.20	2.29	1.58	34.6	50.6
TrajO	$\boldsymbol{e}_P$ (mm)	11.23	8.66	9.15	6.13	18.5	29.2
	$\boldsymbol{e}_O$ (deg)	5.95	5.26	4.78	3.33	19.7	36.7
TrajPO	$\boldsymbol{e}_P$ (mm)	13.58	8.70	11.92	8.06	12.2	7.4
	$\boldsymbol{e}_O$ (deg)	7.19	6.86	3.65	3.43	49.2	50.0

and modelling errors with respect to the position components lying on the image plane (Wang & Wilson, 1992). In general, the peaks on the errors happen when the acceleration is higher, due to the modelling error for the EKF. These errors are partially recovered by the AEKF, in reason of the adaptive law of the state noise covariance matrix $\boldsymbol{Q}$.

The time histories of some of the statistical parameters which are updated on-line in the AEKF are also reported. In particular, the time histories of the elements of the (diagonal) covariance matrix of the state noise are shown in Fig. 6 while the time history of the observation noise for the u and v components is shown in Fig. 7. It can be observed that all the updated parameters keep limited values; moreover, it can be recognized that there exists a correlation between the values of the elements of the $\boldsymbol{Q}$ matrix and the object trajectory. In particular, the peak values of the elements of $\boldsymbol{Q}$ corresponding to the position and orientation components can be related to the peak values of the linear and angular velocity of the object. Analogously, the peak values of the elements of $\boldsymbol{Q}$ corresponding to the linear and angular velocity can be related to the object accelerations.

For completeness, in Fig. 8 the output of the feature selection algorithm for the trajectory *TRajPO* is reported. For each of the 40 feature points, two horizontal lines are considered: a point of the bottom line indicates that the feature point has been classified as visible by the pre-selection algorithm at a particular sample time; a point of

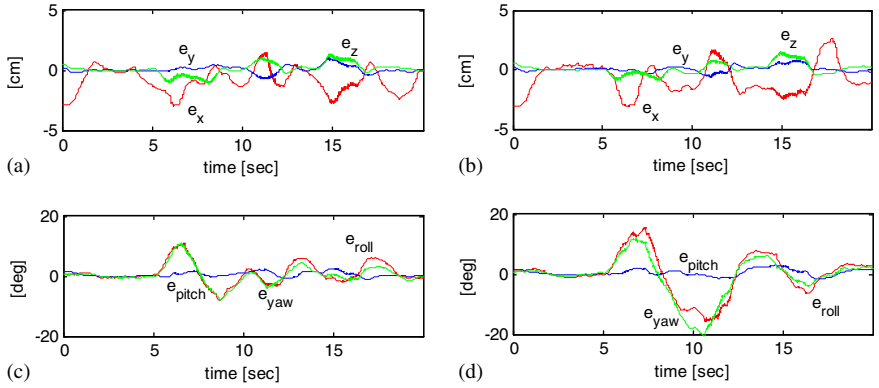

Fig. 5. Time history of the pose estimation errors with AEKF and EKF for the trajectory *TrajPO*. AEKF: (a) position and (c) orientation errors; EKF: (b) position and (d) orientation errors.

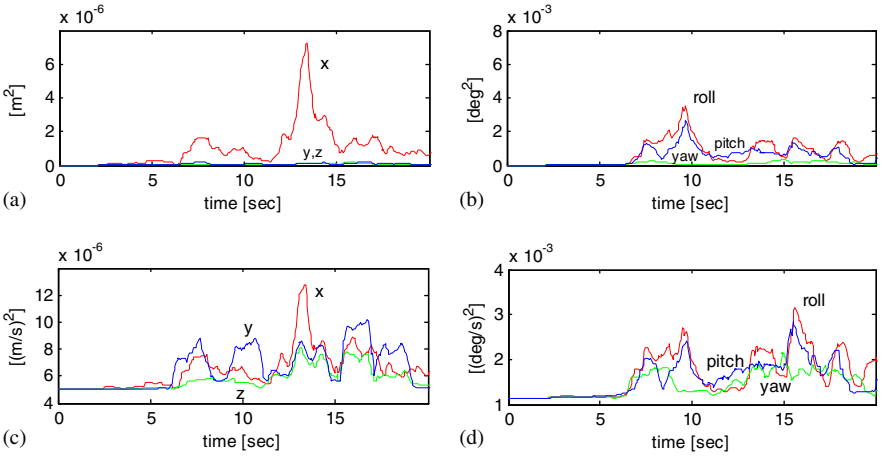

Fig. 6. Time history of the elements of the state noise covariance matrix for the trajectory *TrajPO* in the first case study: (a) position; (b) orientation; (c) linear velocity; (d) rotational velocity components.

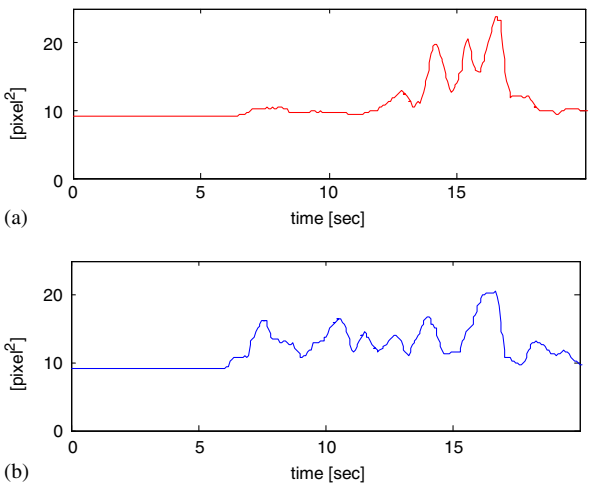

Fig. 7. Time history of the observation noise variance for the trajectory *TrajPO* in the first case study: (a) $\hat{\sigma}^{u2}$; (b) $\hat{\sigma}^{v2}$.

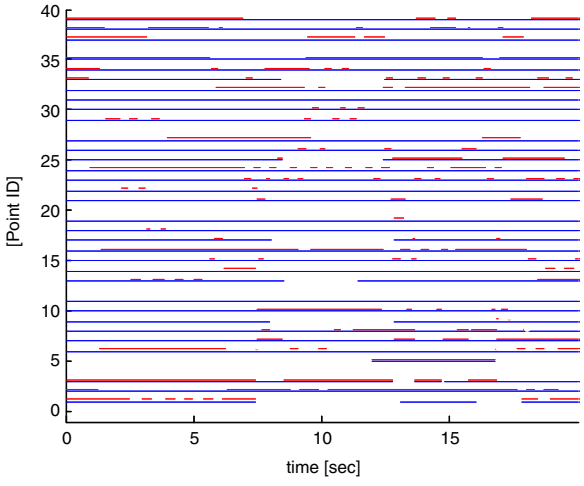

Fig. 8. Visible points and selected points for the trajectory *TrajPO*. For each point, the bottom line indicates when it is visible, the top line indicates when it is selected for feature extraction.

the top line indicates that the visible feature point was chosen by the selection algorithm. Note that nine feature points are selected at each sample time, in order to guarantee at least 5–7 measurements in the case of fault of the extraction algorithm for some of the points. Also, some feature points are hidden during all the motion (points 4, 12, 20, 28, 36, 40) and some points are visible only over partial time intervals (points 1, 3, 5, 9, 13, 17, 25, 33).

Finally, the image seen by the camera during motion is reported in Fig. 9; it is possible to recognize the seven windows effectively used for feature extraction at a given sampling time as well as the small circles inside the windows, corresponding to the measured positions of the feature points.

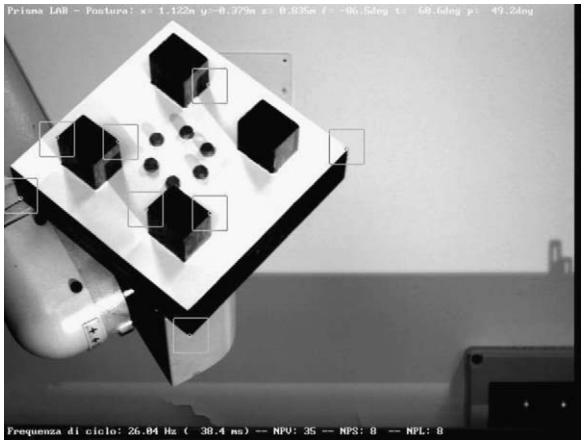

Fig. 9. Image seen by the camera.

Table 2
Comparison of the pose estimation errors for EKF, AEKF-Q, AEKF-R, and AEKF in the second case study

		EKF		Improvement	
		Mean	StD	Mean (%)	StD (%)
EKF	e_P (mm)	13.58	8.70		
	e_O (deg)	7.19	6.86		
AEKF-Q	e_P (mm)	11.94	8.42	12.1	3.2
	e_O (deg)	6.00	4.29	16.6	37.5
AEKF-R	e_P (mm)	12.01	9.64	11.6	−10.8
	e_O (deg)	5.82	5.56	19.1	19.0
AEKF	e_P (mm)	11.92	8.06	12.2	7.4
	e_O (deg)	3.65	3.43	49.2	50.0

7.2. Second case study: updating Q and R separately

In this case study, only the trajectory *TrajPO* has been used. The EKF has been compared to two partial adaptive versions of the EKF: the AEKF-Q, where only matrix $\boldsymbol{Q}$ is updated, and AEKF-R, where only matrix $\boldsymbol{R}$ is updated.

The results are summarized in Table 2 in terms of the mean value and the standard deviation of the norm of the position error $\boldsymbol{e}_P$ and orientation error $\boldsymbol{e}_O$ (the values for the EKF and AEKF are the same reported in Table 1 for the trajectory *TrajPO*). Note that the relative improvement is referred to the values achieved using the EKF. It can be seen that, using the AEKF-Q, good results can be achieved with respect to the EKF, but worse than the complete AEKF, especially for the orientation error. On the other hand, the AEKF-R allows improving all the errors except the standard deviation of the mean value of the position error. This means that the main role to guarantee good tracking performance is played by the matrix $\boldsymbol{Q}$; however, a further improvement can be achieved if both $\boldsymbol{Q}$ and $\boldsymbol{R}$ are updated.

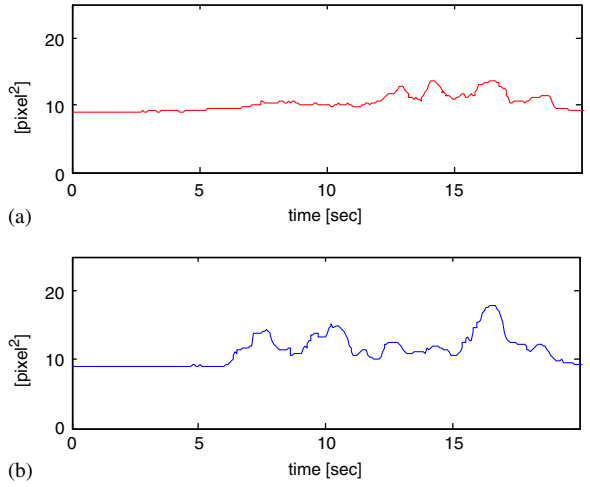

Fig. 11. Time history of the observation noise variance for the AEKF-R in the second case study: (a) $\hat{\sigma}^{u2}$; (b) $\hat{\sigma}^{v2}$.

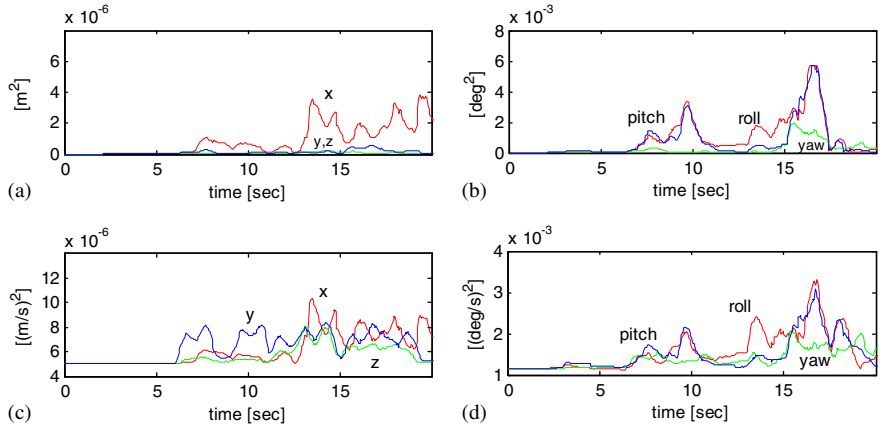

Fig. 10. Time history of the elements of the state noise covariance matrix for the AEKF-Q in the second case study: (a) position; (b) orientation; (c) linear velocity; (d) rotational velocity components.

The time histories of the elements of the (diagonal) covariance matrix of the state noise for AEKF-Q are shown in Fig. 10 while the time history of the observation noise for AEKF-R is shown in Fig. 11.

8. Conclusion

In this paper, an algorithm for the visual motion estimation of the position and orientation of a moving object of known geometry has been proposed. The algorithm fully exploits the prediction capability of the extended Kalman Filter for the pre-selection of the object features to be extracted from the image. An adaptive version of the EKF has been designed, which is capable of automatically tuning the statistics of the observation noise and of the state noise. The experimental results on a single-camera visual system confirm the effectiveness of the AEKF. In fact, the effects on the pose estimation error of the modelling error and of the measurement noise are reduced with respect to the non-adaptive formulation, at the expense of a small increase of computational load. Future work will be devoted to extend the adaptive approach to a multi-camera visual system as well as to the problem of visual estimation of the motion of multiple objects.

Appendix

The computation of the $(m \times 12)$ Jacobian matrices $\boldsymbol{H}_k^u$ and $\boldsymbol{H}_k^v$ in (13) gives

$$\boldsymbol{H}_k^u = \left[\frac{\partial \boldsymbol{g}^u}{\partial x_0} \;\; \mathbf{0} \;\; \frac{\partial \boldsymbol{g}^u}{\partial y_0} \;\; \mathbf{0} \;\; \frac{\partial \boldsymbol{g}^u}{\partial z_0} \;\; \mathbf{0} \;\; \frac{\partial \boldsymbol{g}^u}{\partial \varphi_0} \;\; \mathbf{0} \;\; \frac{\partial \boldsymbol{g}^u}{\partial \vartheta_0} \;\; \mathbf{0} \;\; \frac{\partial \boldsymbol{g}^u}{\partial \psi_0} \;\; \mathbf{0}\right]_k,$$

$$\boldsymbol{H}_k^v = \left[\frac{\partial \boldsymbol{g}^v}{\partial x_0} \;\; \mathbf{0} \;\; \frac{\partial \boldsymbol{g}^v}{\partial y_0} \;\; \mathbf{0} \;\; \frac{\partial \boldsymbol{g}^v}{\partial z_0} \;\; \mathbf{0} \;\; \frac{\partial \boldsymbol{g}^v}{\partial \varphi_0} \;\; \mathbf{0} \;\; \frac{\partial \boldsymbol{g}^v}{\partial \vartheta_0} \;\; \mathbf{0} \;\; \frac{\partial \boldsymbol{g}^v}{\partial \psi_0} \;\; \mathbf{0}\right]_k,$$

where $\mathbf{0}$ is a null $(m \times 1)$ vector corresponding to the partial derivatives of $\boldsymbol{g}^u$ and $\boldsymbol{g}^v$ with respect to the velocity variables, which are null because functions $\boldsymbol{g}^u$ and $\boldsymbol{g}^v$ do not depend on the velocity.

Taking into account the expressions of $\boldsymbol{g}^u$ and $\boldsymbol{g}^v$ in (8), the non-null elements of the Jacobian matrices $\boldsymbol{H}_k^u$ and $\boldsymbol{H}_k^v$ have the form:

$$\frac{\partial}{\partial \lambda}\left(\frac{{}^c x_j}{{}^c z_j}\right) = \left(\frac{\partial\, {}^c x_j}{\partial \lambda}\, {}^c z_j - {}^c x_j \frac{\partial\, {}^c z_j}{\partial \lambda}\right)({}^c z_j)^{-2}, \tag{26a}$$

$$\frac{\partial}{\partial \lambda}\left(\frac{{}^c y_j}{{}^c z_j}\right) = \left(\frac{\partial\, {}^c y_j}{\partial \lambda}\, {}^c z_j - {}^c y_j \frac{\partial\, {}^c z_j}{\partial \lambda}\right)({}^c z_j)^{-2}, \tag{26b}$$

respectively, where $\lambda = x_0, y_0, z_0, \varphi_0, \vartheta_0, \psi_0$ and $j = 1, \ldots, m$.

The partial derivatives on the right-hand side of (26a) and (26b) can be computed as follows.

In view of (2) and (3), the partial derivatives with respect to the components of vector $\boldsymbol{o}_0 = [x_0 \; y_0 \; z_0]^{\mathrm{T}}$ are the elements of the Jacobian matrix

$$\frac{\partial {}^c\boldsymbol{p}_j}{\partial \boldsymbol{o}_0} = \boldsymbol{R}_c^{\mathrm{T}}.$$

In order to express in compact form the partial derivatives with respect to the components of the vector $\boldsymbol{\phi}_0 = [\varphi_0 \; \vartheta_0 \; \psi_0]^{\mathrm{T}}$, it is useful to consider the following equalities

$$\mathrm{d}\boldsymbol{R}_0(\boldsymbol{\phi}_0) = \boldsymbol{S}(\mathrm{d}\boldsymbol{\omega}_0)\boldsymbol{R}_0(\boldsymbol{\phi}_0) = \boldsymbol{R}_0(\boldsymbol{\phi}_0)\boldsymbol{S}(\boldsymbol{R}_0^{\mathrm{T}}(\boldsymbol{\phi}_0)\mathrm{d}\boldsymbol{\omega}_0), \tag{27a}$$

$$\mathrm{d}\boldsymbol{\omega}_0 = \boldsymbol{T}_0(\boldsymbol{\phi}_0)\mathrm{d}\boldsymbol{\phi}_0, \tag{27b}$$

where $\boldsymbol{S}(\cdot)$ is the skew-symmetric matrix operator, $\boldsymbol{\omega}_0$ is the angular velocity of the object frame with respect to the base frame, and the matrices $\boldsymbol{R}_0$ and $\boldsymbol{T}_0$, in the case of Roll, Pitch and Yaw angles, are of the form

$$\boldsymbol{R}_0(\boldsymbol{\phi}_0) = \begin{bmatrix} c_{\varphi_0} c_{\vartheta_0} & c_{\varphi_0} s_{\vartheta_0} s_{\psi_0} - s_{\varphi_0} c_{\psi_0} & c_{\varphi_0} s_{\vartheta_0} c_{\psi_0} + s_{\varphi_0} s_{\psi_0} \\ s_{\varphi_0} c_{\vartheta_0} & s_{\varphi_0} s_{\vartheta_0} s_{\psi_0} + c_{\varphi_0} c_{\psi_0} & s_{\varphi_0} s_{\vartheta_0} c_{\psi_0} - c_{\varphi_0} s_{\psi_0} \\ -s_{\vartheta_0} & c_{\vartheta_0} s_{\psi_0} & c_{\vartheta_0} c_{\psi_0} \end{bmatrix},$$

$$\boldsymbol{T}_0(\boldsymbol{\phi}_0) = \begin{bmatrix} 0 & -s_{\varphi_0} & c_{\varphi_0} c_{\vartheta_0} \\ 0 & c_{\varphi_0} & s_{\varphi_0} c_{\vartheta_0} \\ 1 & 0 & -s_{\vartheta_0} \end{bmatrix},$$

with $c_\alpha = \cos\alpha$ and $s_\alpha = \sin(\alpha)$. By virtue of (27a), (27b), and the properties of the skew-symmetric matrix operator, the following chain of equalities holds

$$\begin{aligned} \mathrm{d}(\boldsymbol{R}_0(\boldsymbol{\phi}_0)^o\boldsymbol{p}_j) &= \mathrm{d}(\boldsymbol{R}_0(\boldsymbol{\phi}_0))^o\boldsymbol{p}_j \\ &= \boldsymbol{R}_0(\boldsymbol{\phi}_0)\boldsymbol{S}(\boldsymbol{R}_0^{\mathrm{T}}(\boldsymbol{\phi}_0)\boldsymbol{T}_0(\boldsymbol{\phi}_0)\mathrm{d}\boldsymbol{\phi}_0)^o\boldsymbol{p}_j \\ &= \boldsymbol{R}_0(\boldsymbol{\phi}_0)\boldsymbol{S}^{\mathrm{T}}({}^o\boldsymbol{p}_j)\boldsymbol{R}_0^{\mathrm{T}}(\boldsymbol{\phi}_0)\boldsymbol{T}_0(\boldsymbol{\phi}_0)\,\mathrm{d}\boldsymbol{\phi}_0 \\ &= \boldsymbol{S}^{\mathrm{T}}(\boldsymbol{R}_0(\boldsymbol{\phi}_0)^o\boldsymbol{p}_j)\boldsymbol{T}_0(\boldsymbol{\phi}_0)\,\mathrm{d}\boldsymbol{\phi}_0, \end{aligned}$$

hence

$$\frac{\partial \boldsymbol{R}_0(\boldsymbol{\phi}_0)}{\partial \boldsymbol{\phi}_0}{}^o\boldsymbol{p}_j = \boldsymbol{S}^{\mathrm{T}}(\boldsymbol{R}_0(\boldsymbol{\phi}_0)^o\boldsymbol{p}_j)\boldsymbol{T}_0(\boldsymbol{\phi}_0). \tag{28}$$

At this point, by virtue of (2), (3) and (28), the following equality holds

$$\begin{aligned} \frac{\partial {}^c\boldsymbol{p}_j}{\partial \boldsymbol{\phi}_0} &= \boldsymbol{R}_c^{\mathrm{T}} \frac{\partial \boldsymbol{R}_0(\boldsymbol{\phi}_0)}{\partial \boldsymbol{\phi}_0}{}^o\boldsymbol{p}_j \\ &= \boldsymbol{R}_c^{\mathrm{T}}\boldsymbol{S}^{\mathrm{T}}(\boldsymbol{R}_0(\boldsymbol{\phi}_0)^o\boldsymbol{p}_j)\boldsymbol{T}_0(\boldsymbol{\phi}_0). \end{aligned}$$

References

Bai, M., Zhou, D. H., & Schwarz, H. (1998). Adaptive augmented state feedback control for an experimental planar two-link flexible manipulator. *IEEE Transactions on Robotics and Automation*, *14*, 894–901.

Bradshaw, K. J., Reid, I. D., & Murray, D. W. (1997). The active recovery of 3D motion trajectories and their use in prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(3), 219–233.

Broida, T., & Chellappa, R. (1986). Estimation of object motion parameters from noisy images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *1*, 90–99.

Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *8*, 679–698.

Chiuso, A., Favaro, P., Jin, H., & Soatto, S. (2002). Structure from motion casually integrated over time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(4), 1–13.

Corke, P. I. (1996). *Visual control of robots: High performance visual servoing*. New York: Wiley.

Corke, P. I., & Hutchinson, S. A. (2001). A new partitioned approach to image-based visual servo control. *IEEE Transactions on Robotics and Automation*, *17*, 507–515.

Drummond, T., & Cipolla, R. (2002). Real-time visual tracking of complex structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*, 932–946.

Espiau, B., Chaumette, F., & Rives, P. (1992). A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation*, *8*, 313–326.

Feddema, J. T., Lee, C. S. G., & Mitchell, O. R. (1991). Weighted selection of image features for resolved rate visual feedback control. *IEEE Transactions on Robotics and Automation*, *7*, 31–47.

Ficocelli, M., & Janabi-Sharifi, F. (2001). Adaptive filtering for pose estimation in visual servoing. *Proceedings of the 2001 IEEE/RSJ international conference on intelligent robots and systems* (pp. 19–24).

Girgis, A. A., & Peterson, W. L. (1990). Adaptive estimation of power system frequency deviation and its rate of change for calculating sudden power system overloads. *IEEE Transactions on Power Delivery*, *5*, 585–594.

Harris, C. (1992). Tracking with rigid models. In A. Blake, & A. Yuille (Eds.), *Active vision* (pp. 57–73). Cambridge, MA: MIT Press.

Hashimoto, K. (1993). *Visual servoing: Real time control of robot manipulators based on visual sensory feedback*. Singapore: World Scientific.

Hutchinson, S., Hager, G. D., & Corke, P. I. (1996). A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, *12*, 651–670.

Janabi-Sharifi, F., & Wilson, W. J. (1997). Automatic selection of image features for visual servoing. *IEEE Transactions on Robotics and Automation*, *13*, 890–903.

Jetto, L., Longhi, S., & Venturini, G. (1999). Development and experimental validation of an adaptive extended Kalman filter for the localization of a mobile robot. *IEEE Transactions on Robotics and Automation*, *15*, 219–229.

Jetto, L., Longhi, S., & Vitali, D. (1999). Localization of a wheeled mobile robot by sensor data fusion based on a fuzzy logic adapted Kalman filter. *Control Engineering Practice*, *7*, 763–771.

Kriegman, D., Hager, G. D., & Morse, S. (Eds.), (1998). *The confluence of vision and control*. New York: Springer.

Lee, S., & Kay, Y. (1990). An accurate estimation of 3-D position and orientation of a moving object for robot stereo vision: Kalman filter approach. *Proceedings of 1990 IEEE international conference on robotics and automation* (pp. 414–419).

Lewis, F. L. (1986). *Optimal estimation with an introduction to stochastic control theory*. New York: Wiley.

Lippiello, V., Siciliano, B., & Villani, L. (2002). A new method of image features pre-selection for real-time pose estimation based on Kalman filter. *Proceedings of 2002 IEEE/RSJ international conference on intelligent robots and systems* (pp. 372–377).

Lippiello, V., & Villani, L. (2003). Managing redundant visual measurements for accurate pose tracking. *Robotica*, *21*, 511–519.

Malis, E., Chaumette, F., & Boudet, S. (1999). 2 1/2 D visual servoing. *IEEE Transactions on Robotics and Automation*, *15*, 234–246.

Myers, K. A., & Tapley, B. D. (1976). Adaptive sequential estimation with unknown noise statistics. *IEEE Transactions on Automatic Control*, *21*, 520–523.

Philip, J. (1991). Estimation of three dimensional motion of rigid objects form noisy observations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *13*(1), 61–66.

Tarabanis, K., Tsai, R. Y., & Kaul, A. (1996). Computing occlusion-free viewpoints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *18*, 279–292.

Thuilot, B., Martinet, P., Cordesses, L., & Gallice, J. (2002). Position based visual servoing: Keeping the object on the field of vision. *Proceedings of 2002 IEEE International Conference on Robotics and Automation* (pp. 1624–1629).

Vincze, M., & Hager, G. D. (Eds.), (2000). *Robust vision for vision-based control of motion*. Piscataway, NJ: IEEE Press.

Wang, J., & Wilson, W. J. (1992). 3D relative position and orientation estimation using Kalman filter for robot control. *Proceedings of 1992 IEEE international conference on robotics and automation* (pp. 2638–2645).

Weng, J., Cohen, P., & Herniou, M. (1992). Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *14*, 965–980.

Wilson, W. J., Hulls, C. W., & Bell, G. (1996). Relative end-effector control using cartesian position based visual servoing. *IEEE Transactions on Robotics and Automation*, *12*, 684–696.

Wira, P., & Urban, J. P., 2000. A new adaptive Kalman filter applied to visual servoing tasks. *Proceedings of fourth International conference on knowledge-based intelligent engineering systems and allied technologies* (pp. 267–270).