Object Recognition and Tracking for Remote Video Surveillance

Gian Luca Foresti, *Member, IEEE*

***Abstract*—In this paper, a system for real-time object recognition and tracking for remote video surveillance is presented. In order to meet real-time requirements, a unique feature, i.e., the statistical morphological skeleton, which achieves low computational complexity, accuracy of localization, and noise robustness has been considered for both object recognition and tracking. Recognition is obtained by comparing an analytical approximation of the skeleton function extracted from the analyzed image with that obtained from model objects stored into a database. Tracking is performed by applying an extended Kalman filter to a set of observable quantities derived from the detected skeleton and other geometric characteristics of the moving object. Several experiments are shown to illustrate the validity of the proposed method and to demonstrate its usefulness in video-based applications.**

***Index Terms*—Extended Kalman filter, image processing, object recognition, object tracking, safety monitoring, video surveillance.**

I. Introduction

RECOGNITION and tracking of three-dimensional (3-D) objects from image sequences recorded by a stationary camera is a basic task for several applications of Computer Vision, e.g., road traffic control [1], airport safety [2], surveillance systems [3]–[4], etc. Recognition can be used to give an interpretation of a 3-D scene or in discriminating among different objects interacting with it. Tracking is used to detect moving objects and to pursue the objects of interest by estimating their motion parameters, e.g., trajectory, speed, orientation, etc. Several vision systems have been focused on object recognition [5]–[9] or object tracking [10]–[12], but few systems have been developed to take into account both object recognition and tracking tasks.

The problem of object recognition is one of the ubiquitous problems of Computer Vision and appears in a variety of guises. Given sensory data about a 3-D scene and given a set of object models, the goal of object recognition is to find instances of these objects in the scene. This simple statement contains a large variety of key issues, e.g., sensor data acquisition, representation of object models, matching data, and models, which have been analyzed in the last years with different approaches. Some techniques rely on the use of known geometric models of the object being recognized. Their goal is to estimate the location and the orientation of the objects of interest, e.g., acronym [5] or the system proposed by Fua [6] which employ a parameterized and a generic model, respectively. A second category involves no dependency on stored geometric models. Recognition is attempted on the basis of the cues besides shape, such as size, location, appearance, and context. MSYS [7] and Condor [8] are examples of such kind of systems. An exhaustive survey of object recognition techniques is given in [9].

Tracking techniques are divided into two categories: recognition-based tracking [10], [11] and motion-based tracking [12]. Recognition-based tracking concerns the recognition of the object in successive images and the extraction of its position. The main advantage of this tracking method is that it can be achieved in three dimensions, and that the object translation and rotation can be estimated. The disadvantage is that only recognized objects can be tracked, and, consequently, the tracking performances are limited by the high computational complexity of the recognition method. Motion-based tracking systems rely entirely on motion parameter estimation to detect the object. They have the advantage of being able to track any moving object regardless of size or shape.

Some recent works have addressed the problem of developing complete vision systems for both object recognition and tracking in order to obtain a rough scene understanding [13]–[15]. However, recognition and tracking tasks are not integrated in a common spatio-temporal domain so that occlusions and noise can generate false object appearance in the scene. Moreover, the tracking and the recognition are based on different kind of features so that the computational complexity of the whole system becomes very high. A recognition-based system for object tracking in traffic scenes has been proposed by Koller *et al.* [13]. This system uses *a priori* knowledge about the shapes and the motion of vehicles to be tracked and recognized in a scene. In particular, a parameterized vehicle model which takes into account also shadow edges allows the system to work under complex lighting conditions and with a small effective field of view. The main limitations of this method are computational complexity (due especially to the line extraction and matching processes), dependency by initial matches and object pose estimates, and false matching combinations. Another interesting system for recognition and tracking multiple vehicles in road traffic scenes has been developed by Malik *et al.* [14]. This system addresses the problem of occlusions in tracking multiple 3-D objects in a known environment by employing a contour tracker algorithm based on intensity and motion boundaries. The position and the motion of contours, represented by closed cubic splines, are estimated

Manuscript received October 14, 1997; revised April 22, 1999. This paper was recommended by Associate Editor T. Sikora.

The author is with the Department of Mathematics and Computer Science (DIMI), University of Udine, 33100 Udine, Italy (e-mail: foresti@dimi.uniud.it).

Publisher Item Identifier S 1051-8215(99)08180-X.

by means of linear Kalman filters. However, the recognition capability of the system is limited to two-dimensional (2-D) object shapes on the image plane to solve the problem of tracking adjacent or overlapped vehicles. The system, which has been extensively tested in various traffic situations, reaches good performances in the object detection and tracking tasks, but presents some limitations in the recognition phase (false object appearance), due essentially to the simplicity of the used 2-D object models. A visual surveillance system for remote monitoring of railway level crossings has been proposed by Foresti [15]. The system is able to detect, localize, track, and classify multiple objects moving in the surveilled area. In order to satisfy real-time constraints, object classification and object tracking are performed independently on different and simple features, i.e., the specstrum [16] and the center of mass of the object on the image plane. Several tests demonstrate how the system reduces both false and missed alarms, even with the limitation that objects are moving with similar trajectories.

In this paper, a common framework for real-time object recognition and tracking for remote video surveillance is described. The main novelty of the paper is that, a unique feature, i.e., the statistical morphological skeleton (SMS) [17], is used in a parallel and fast way for both object recognition and tracking. The SMS which is a shape descriptor operator achieving low computational complexity, accuracy of localization and noise robustness, is extended here to real-time applications and outdoor scenes. Recognition is obtained by comparing an analytical approximation of the skeleton function extracted from the analyzed image with that obtained from model objects stored into a database [18]. Tracking is performed by applying an extended Kalman filter to a set of observable quantities derived from the detected SMS and other geometric characteristics of the moving object.

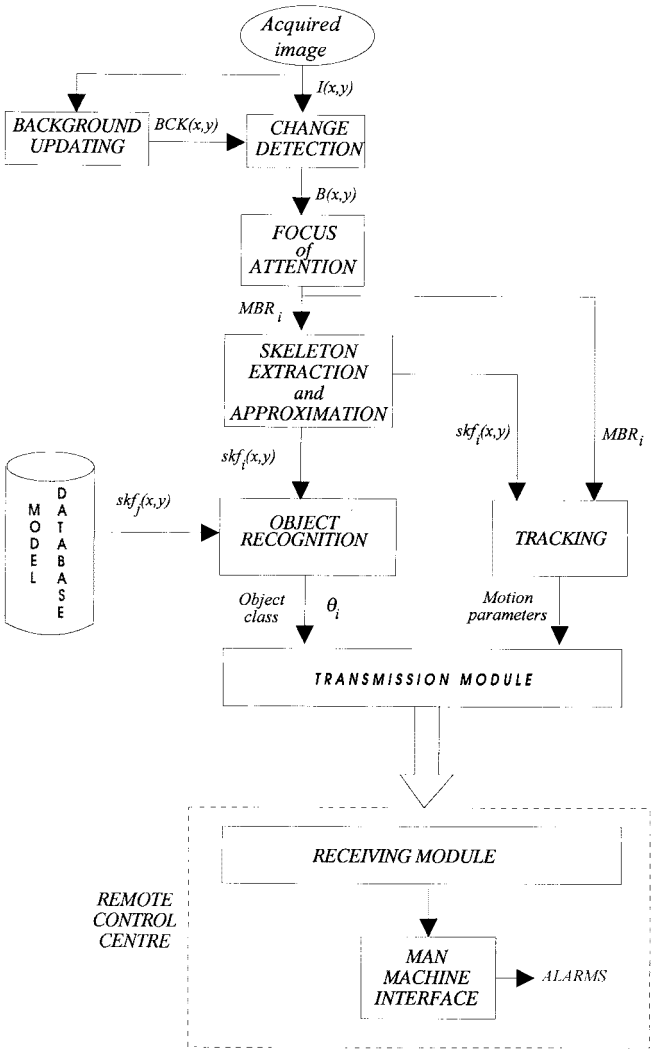


Fig. 1. General system architecture.

II. System Description

The general scheme of the system is shown in Fig. 1. A stationary charge-coupled device camera is used to acquire image sequences of a surveilled scene. A change detection (CD) module[19] is applied to each frame $I(x,y)$ of the input image sequence and it makes out a binary image $B(x,y)$ where each blob represents a possible object (e.g., a car, a truck, etc.) moving in the scene. At first, the absolute difference $D(x,y)$ between the current image $I(x,y)$ and a background image $BCK(x,y)$ is computed every t_k time instants

$$D(x,y) = |I(x,y) - BCK(x,y)| \quad \forall (x,y) \in [1,N] \times [1,M] \tag{1}$$

where $N \times M$ is the image size. Then, a hysteresis function with two thresholds, THR_{in} and THR_{out}, and two states, *background* and *object*, is applied to the $D(x,y)$ image to establish whenever a point belongs to the background or to a moving object, i.e.,

$$B(x,y) = \begin{cases} 0, & \text{if state} = \textit{object} \text{ and } D(x,y) < THR_{\text{out}} \\ 1, & \text{if state} = \textit{background} \text{ and } D(x,y) > THR_{\text{in}}. \end{cases} \tag{2}$$

At the end of the process, a binary image $B(x,y)$ where changing pixels are set to one and background pixels are set to zero, is obtained. A background updating procedure is applied to estimate significant changes of the background scene [19]. A Kalman filter is applied to each image pixel $\boldsymbol{p} = (x,y)$ of the input image to adaptively predict a background estimate at each time instant. The applied Kalman filter is characterized by the following equations:

$$S(i+1,\boldsymbol{p}) = S(i,\boldsymbol{p}) + \mu(i) \quad \text{dynamic model} \tag{3a}$$

$$I(i,\boldsymbol{p}) = S(i,\boldsymbol{p}) + \eta(i,\boldsymbol{p}) \quad \text{measure model} \tag{3b}$$

where $S(i,\boldsymbol{p})$ represents the gray level of the background image point $\boldsymbol{p}$ at the ith frame, $S(i+1,\boldsymbol{p})$ represents an estimate of the same quantity at the successive frame $(i+1)$, $\mu(i)$ is an estimate of the system error, $I(i,\boldsymbol{p})$ represents the gray level of the current image point $\boldsymbol{p}$, and $\eta(i,\boldsymbol{p})$ is the estimate of the noise affecting the input image, i.e., measure system error. The main advantage of this approach is that it takes into account both slow scene variations, due to illumination changing, and sudden variations, due to the entrance in the scene of new objects.

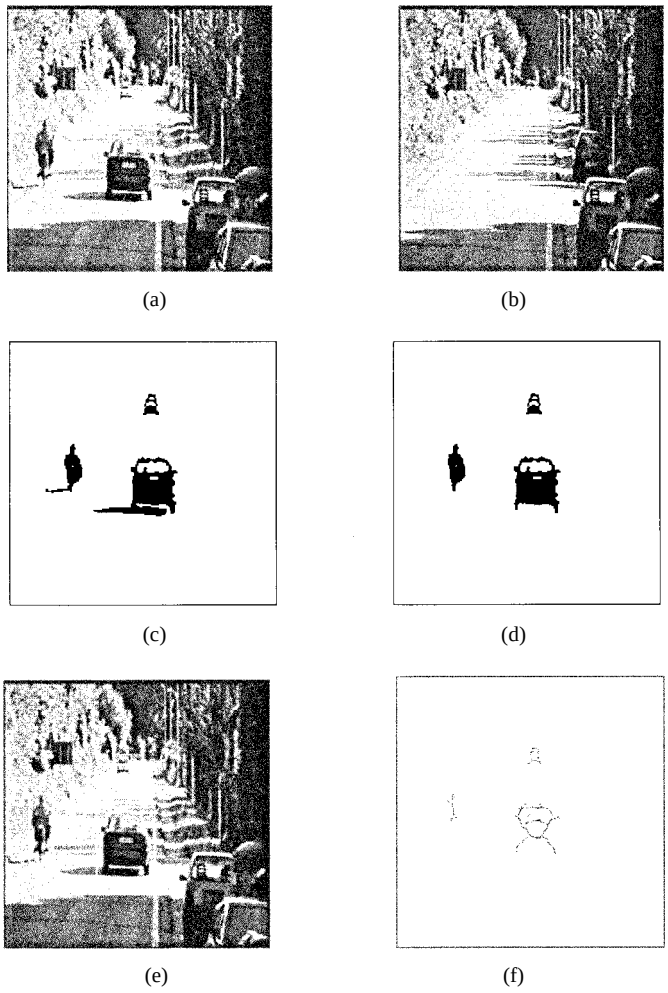


Fig. 2. (a) Real image representing a country road characterized by multiple vehicles and shadows, (b) background image, (c) $B(x,y)$ image obtained with $THR_{\text{in}} = 20$ and $THR_{\text{out}} = 40$, (d) $B(x,y)$ image, after shadow elimination, (e) MBR's detected by th FA module, and (f) projection on the plane (x,y) of the detected skeleton functions.

Fig. 2(a) shows a real image (chosen as a running example) representing a country road characterized by multiple vehicles and shadows. Fig. 2(b) and 2(c) show the background and the obtained $B(x,y)$ image, respectively. A shadow detection procedure is applied to reduce the shadow effects from the object blobs [20] [Fig. 2(d)].

Then, a focus of attention (FA) module provides the minimum bounding rectangle (MBR) of each blob region and labels them as different candidate moving objects. The FA module works on two levels. At the first level, two binary morphological operators, e.g., erosion and dilation, are applied to reduce the noise and to obtain a binary image characterized by uniform and compact regions. A 3×3 square structuring element is used. Fig. 2(e) shows the output of the FA module applied to the image of the running example. In general, noise effects can produce on the image plane false blobs which can be eliminated afterwards by the tracking module (by integrating ground plane hypothesis and object motion constraints).

Then, the statistical morphological skeleton is extracted from each MBR and the related skeleton function $skf(x,y)$ is approximated by means of B splines [21]. Fig. 2(f) shows the projection of the plane (x,y) of the $skf(x,y)$ extracted from the MBR's in Fig. 2(e). Thanks to the invariance of B-splines, it is possible to compare the morphological information contained in the skeleton functions, $skfi(x,y)$ and $skfj(x,y)$, related to two different object shapes. To this aim, a recognition module classifies unknown objects and estimates their 3-D orientation θ_i by comparing their approximated skeleton functions with those of model objects stored into a database. Moreover, a tracking module which uses as input geometric information about the MBR's and the skeleton function $skf(x,y)$ is applied to estimate the motion parameters of the detected objects. Finally, information about the object class and its position in the scene is transmitted to a remote control center and displayed to a human operator.

III. Feature Extraction

The growing number of recent studies on feature extraction shows that it is difficult to select features achieving accuracy of localization, consistency of detection, and low computational complexity. Morphological skeleton (MS) [22], [23] is an information-preserving shape-descriptor which shows all these characteristics, but it is noise dependent [24]. To this purpose, a new noise robust shape descriptor, i.e., the statistical morphological skeleton (SMS) [17], has been considered as a common feature for object recognition and tracking.

A. Statistical Morphological Skeleton

The idea of transforming an image to a skeleton has been introduced in 1961 by Blum [22], who called it medial axis transform. Subsequently, a mathematical theory for the skeleton of continuous images and for discrete images has been developed [23]. More recently, Maragos and Shafer have proposed a new algorithm for fast calculation of MS [24].

The statistical morphological skeleton of an object can be obtained by applying successive binary statistical morphology (BSM) transformations to the object and to the sequence of shrunk images that can be derived from it [17]. BSM operators which are characterized by binary inputs and binary outputs result from a generalization of morphological ones, in that they allow one to take into account noise effects [25].

Let us define $M_B^1(m) = card\{x \in (B + m \cap X)\}$ as the number of elements of a binary shape $X = \{(i,j)\colon (i,j) \in I\}$ in the area of the structuring element B shifted to position m, where $card(A)$ is the cardinality of the set A and $I = \{(i,j)\colon i = 1, \ldots, M \; j = 1, \ldots, N\}$ defines the image lattice. Analogously, the number of zeros in the same area is denoted as $M_B^0(m) = card\{x \in (B + m \cap X^c)\}$, where $X^c = \{(i,j)\colon (i,j) \in I, (i,j) \notin X\}$. It holds that

$$M_B^1(m) + M_B^0(m) = card(B + m) = card(B) = N_B. \quad (4)$$

Binary statistical erosion (BSE) and binary statistical dilation (BSD) operators are defined as

$$X \ominus_\beta B = \{m\colon E_m > \theta, \beta \leq 0, m \in I\} \quad \text{(BSE)} \quad (5a)$$

$$X \oplus_\beta B = \{m\colon H_m > \theta, \beta \geq 0, m \in I\} \quad \text{(BSD)} \quad (5b)$$

where $\theta \in (0, 1)$ is a threshold related to the expected costs of output configurations [22], and

$$E_m = \frac{M_B^1(m)\exp(-\beta)}{N_B + M_B^1(m)(\exp(-\beta)-1)} \tag{6a}$$

$$H_m = \frac{M_B^1(m)\exp(\beta)}{N_B + M_B^1(m)(\exp(\beta)-1)}. \tag{6b}$$

It is possible to perform SMS extraction by selecting different BSM operators, i.e., choosing different β values (analyzing the object shape at different degrees of resolution). An appropriate scheduling of β values through iterations makes it possible to obtain an increased stability to noise, as compared with solutions that keep this parameter fixed during shape analysis.

The method introduced in [17] for extracting the SMS from synthetic binary images corrupted by noise is extended here to work on binary images extracted from real scenes. This extension needs to avoid recording of wrong shape information related to false border details due both to noise affecting the image acquisition process and to errors produced by the CD process. Let $\Gamma_{n,\beta_n}(X) = (X \ominus_{\beta_n} B) \oplus_{\beta_n} B$ be the statistical opening operator at stage n. The statistical skeleton $\mathrm{SMS}(X)$ of a binary image X can be obtained according to the following algorithm:

1) *Initialization:* $n = 0$, $S_{-1}(X) = X$.
2) *Iterations of the following steps:*
   a) $S_n(X) = (S_{n-1}(X) \ominus_{\beta_n} B)$
   b) ***if*** $S_n(X) = \emptyset$ ***then*** $R_{Nx}(X) = S_{Nx-1}(X)$; ***Stop;***
   c) $\Gamma_{n,\beta n}(X) = S_n(X) \oplus_{\beta_n} B$
   d) $R_n(X) = S_{n-1}(X) - \Gamma_{n,\beta_n}(X)$
   e) $n = n + 1$; $\beta_n = \ln$ (*gain* $\times$ n) + *offset*; *if* $n > N_x$ *then* ***Stop, else goto 1***

where $-$ is the set difference operator, $N_x = \max\{n\colon S_n(X) \neq \emptyset\}$, and n represents the current step. The SMS is provided as the combination of representations at intermediate steps, i.e., $\mathrm{SMS}(X) = \cup_{i=1}^{N_x} R_n(X)$. Step (4) defines the shape extraction measure by considering only points which are in $S_{n-1}(X)$ and not in $\Gamma_{n,\beta_n}(X)$ are considered. This choice may cause loss of some shape information, despite avoiding recording of wrong shape information related to not existent details. Step (5) defines a logarithmic scheduling of the β parameter, chosen by analogy with simulated annealing methods [26], that corresponds to the selection of different rank-order filters at successive iterations of the method [27]. This process of gradually switching is necessary because the process of separating noise from contour information is critical and must be performed slowly. *Gain* and *offset* are chosen in order to obtain $\beta \cong 0$ for low n values. The so obtained SMS points are characterized by a greater connectivity and the related skeleton function has a more continuous behavior.

Fig. 3(a) shows a real image representing a truck coming in the direction of the camera and Fig. 3(b) shows the binary image $B(x, y)$. Fig. 3(c) shows the intermediate representations $R_n(X)$ at successive iterations. In this case, for $\beta = 0$, no change occurs after the first iteration, due to the presence of fixed points; for increasing β values, the convergence speed increases toward an empty set. Fig. 3(d) shows the image in Fig. 3(b) corrupted by impulsive noise with percentages equal, respectively, to 10, 15, and 25% from left to right. Fig. 3(e) and (f) show, respectively, the MS and the SMS with logarithmic scheduling used to regulate β_n through various iterations such that $\beta_0 = 0$ and $\beta_n \to \infty$ for $n \to \infty$. The SMS remains quite stable up to 25% of noise added, while the MS ($\beta \to \infty$) lacks the connectivity already for 10% of noise.

B. Statistical Morphological Skeleton Approximation

Let $skf_i(x, y)$ be the skeleton function related to the ith detected object which associates with each skeleton point the iteration n at which the point itself has been detected. Fig. 4(a) shows the function $skf_i(x, y)$ obtained for each blob extracted from the real image in Fig. 2. In order to simplify and to increase the speed of the object recognition and tracking phases, the function $skf_i(x, y)$ is approximated by four nonuniform rational B-splines (NURBS) [21] as

$$skf_i(x, y) = \bigcup_{j=1}^{4} \boldsymbol{Q}_{ij}(t) \quad \text{with}$$
$$\boldsymbol{Q}_{ij}(t) = [x_{ij}(t), y_{ij}(t), z_{ij}(t)] \tag{7a}$$

where $x_{ij}(t)$, $y_{ij}(t)$, and $z_{ij}(t)$ take on the form given by the following equation, expressed as a function of the generic coordinate s ($s = x$ or y or z):

$$s_{ij}(t) = a_{s_{ij}}(t - t_j)^3 + b_{s_{ij}}(t - t_j)^2 + c_{s_{ij}}(t - t_j) + d_{s_{ij}}$$
$$t \in [t_j, t_{j+1}] \quad (j = 1, \ldots, 4). \tag{7b}$$

Fig. 4(b) shows the approximation of the $skf_i(x, y)$ functions in Fig. 4(a). Let $\boldsymbol{v}_{s_{ij}} = [a_{s_{ij}}, b_{s_{ij}}, c_{s_{ij}}, d_{s_{ij}}]$ be the vector of coefficients of $s_{ij}(t)$. Each coefficient vector is then normalized, i.e., $\boldsymbol{v}_{s_{ij}} = [\boldsymbol{v}_{s_{ij}}/||\boldsymbol{v}_{s_{ij}}||]$ $(j = 1, \ldots, 4)$.

IV. Object Recognition and 3-D Orientation Estimation

Thanks to the invariance properties of NURBS functions with respect to affine transformations [21], it is possible to compare the morphological information contained in the skeleton functions related to two different object shapes by means of the respective normalized vectors of coefficients $\boldsymbol{V}_{s_i} = [\boldsymbol{V}_{s_{i1}}, \boldsymbol{V}_{s_{i2}}, \boldsymbol{V}_{s_{i3}}, \boldsymbol{V}_{s_{i4}}]$. To this aim, the unknown object and its related 3-D orientation can be determined through a comparison of the vector $\boldsymbol{V}_{s_i}$ related to the ith detected object, with $K \times H$ vectors, $(\boldsymbol{V}^h_{s_1}, \ldots, \boldsymbol{V}^h_{s_K})$, $k = 1, \ldots, K$, $h = 1, \ldots, H$, related to the known hth object model with orientation k. Each vector $\boldsymbol{V}^h_{s_k}$ is computed off-line by starting from a synthetic binary image representing the hth object model with orientation k and stored in a database. In several real applications [1–14], H belongs to the range [1, 50] and K depends on the orientation estimation accuracy requested, e.g., $K = 360$ for an accuracy of one degree. The angle between the camera and the ground plane is fixed, i.e., a stationary camera is used. Fig. 5 show the 2-

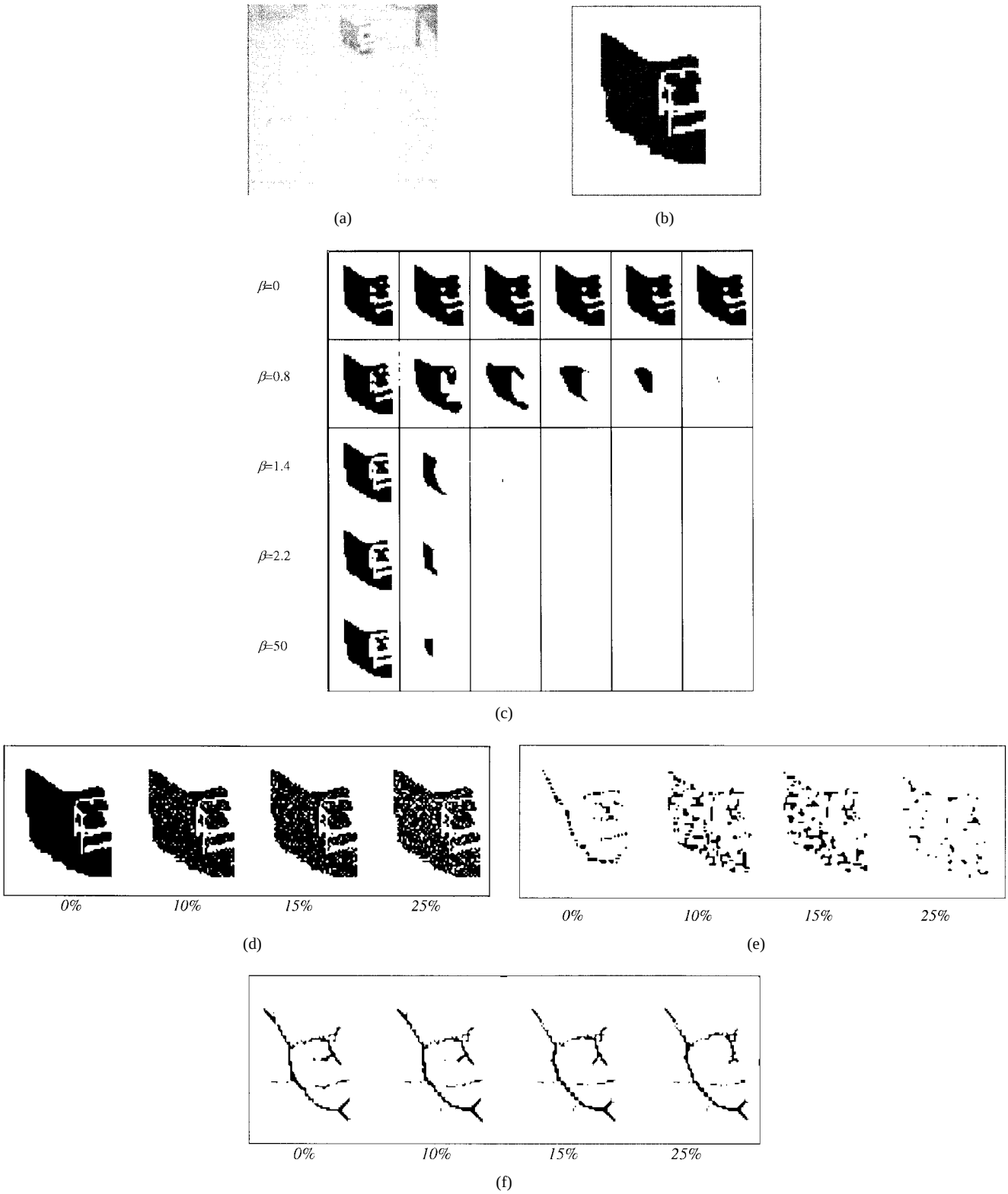
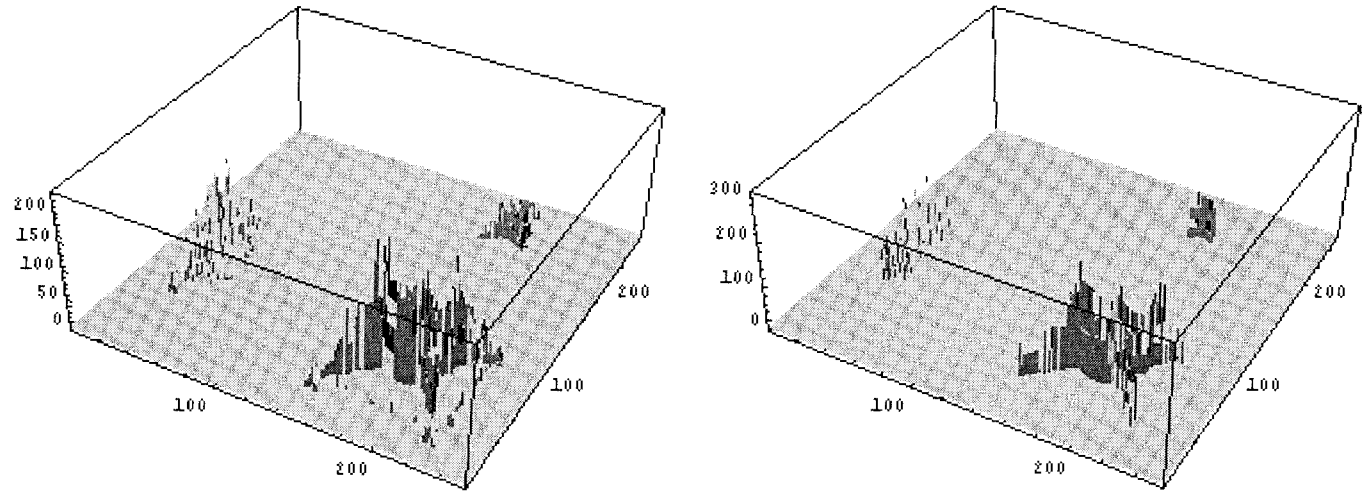


Fig. 3. (a) Real image representing a truck moving in the direction of the camera, (b) $B(x, y)$ image, (c) intermediate representations $R_n(X)$ for different β values. (d) Binary test images obtained by adding to the $B(x, y)$ image impulsive noise with different percentages (10, 15, and 25% from left to right); (e) morphological skeleton and (f) SMS obtained by applying a logarithmic scheduling of the β parameter.

D shapes representing different views of 3-D synthetic object models.

The matching process to estimate the unknown object class and the object orientation is performed as follows. For each ith detected object, a function, $J_i = \{j_{ihk}\}, k = 1, \ldots, K, h = 1, \ldots, H$, is computed, where $j_{ihk} = \alpha_s \Delta s_{ihk}$. In particular, $\Delta s_{ihk} = ||\boldsymbol{V}_{s_i} - \boldsymbol{v}^h_{s_k}||$, where the norm $||\ ||$ represents the Euclidean distance between two vectors. The coefficient α_s



(a) (b)

Fig. 4. (a) Skeleton function $skf_i(x, y)$ and (b) relative approximation obtained for the blobs extracted from the real image in Fig. 2.

is obtained by minimizing the cost function $C(\alpha_s)$ on a long training sequence

$$C(\alpha_s) = \sum_{h=1}^{H} \left\{ \sum_{m \in \Omega_h} |J_m - U_m|^2 \right\} = \sum_{h=1}^{H} \left\{ \sum_{m \in \Omega_h} \left\{ \sum_{k=1}^{K} [j_{mhk} - u_{mhk}]^2 \right\} \right\} \quad (8)$$

where Ω_h is the training set (images representing the same object model in different poses) for the hth class and

$$u_{mhk} = \begin{cases} 0, & \text{if } k_m = k^* \text{ and } h_m = h^* \\ U > 0, & \text{otherwise} \end{cases}$$

is the kth element of the target vector U_m (h^* and k^* represent, respectively, the class and orientation for the mth object of the training set). As the cost function $C(\alpha_s)$ is composed only by positive terms, the minimum can be determined by a gradient descent algorithm with a variable step size [28]. The unknown object class h_m and object orientation k_m are determined by solving the following equation:

$$(h_{mi}, k_m) = \underset{h \in [1,\ldots,H], k \in [1,\ldots,K]}{\operatorname{argmin}} \{j_{mhk}\}. \quad (9)$$

Finally, the detected object is recognized as belonging to the class h_m if and only if it is assigned to the same class for at least three consecutive frames.

V. Object Tracking

The goal of the tracking module is to estimate the object position and trajectory at each image frame. An extended Kalman filter (EKF) [29] is applied to solve the object tracking problem. To this purpose, the tracking module which uses data coming from low level modules is characterized by four principal steps: (A) selection of the quantities of interest (QI's); (B) selection of the measurable quantities (QM's), (C) dynamic model formulation for the QI's, and (D) application of an EKF for the estimation of the QI's.

A. Selection of the QI's

Let (X, Y, Z) be the 3-D reference system of the observed scene [Fig. 6(a)]. Let (X_c, Y_c, Z_c) be the camera reference system whose origin is placed in the point (0, h_c, 0). The camera is characterized by a tilt angle α, i.e., angle between the optical axis Z_c and the plane $Y = 0$, and it has been calibrated by means of the algorithm developed by Tsai [30]. Let (x, y) be the image coordinate system [Fig. 6(b) and (c)]. QI's are variables which compose the status vector $\boldsymbol{\Phi_i}$. In particular, eight quantities have been selected to describe the trajectory on the ground plane XZ of the barycenter $\boldsymbol{B_i}$ of the parallelepiped bounding the object: X_{i-1}, X_i, $\dot{X}_{i-1}$, $\dot{X}_i$ and Z_{i-1}, Z_i, $\dot{Z}_{i-1}$, $\dot{Z}_i$. Other four quantities have been selected to represent the object size and orientation: height H, width W, and length D, and 3-D orientation θ_i, with respect to one of its principal axes [Fig. 6(d)].

B. Selection of the QM's

In order to complete the dynamic system, eight measurable quantities computed by the FA and the SMS extraction modules have been considered. As a nonlinear relation exists between the QI's and the QM's, i.e., $\boldsymbol{y}_i = h(\boldsymbol{\Phi}_i) + \boldsymbol{\eta}_i$, where $\boldsymbol{\eta}_i = N(0, \boldsymbol{R}_i)$ is a Gaussian noise with zero means and covariance matrix $\boldsymbol{R}_i$, the EKF model requires one to linearize the system by approximating the function h with Taylor's series expansions, retaining only first-order terms

$$\boldsymbol{y}_i = \boldsymbol{H}_i(\boldsymbol{\Phi}_i^-)\boldsymbol{\Phi}_i^- + \boldsymbol{\eta}_i \quad (10)$$

where $\boldsymbol{\Phi}_i^-$ is the status vector predicted by the Kalman filter and $\boldsymbol{H}_i(\boldsymbol{\Phi}_t) = (d/d\boldsymbol{\Phi}_i)h(\boldsymbol{\Phi}_i)$.

The first measure is represented by the coordinate x_i of the point $\boldsymbol{b}_i$ which is the projection on the image plane of the barycenter $\boldsymbol{B}_i$ of the parallelepiped bounding the object. The second measure is represented by the x_i coordinate taken at the previous time instant x_{i-1} and the third one is represented by the displacement $\Delta x = (x_i x_{i-1})$. The relation between each coordinate x_i and the QI's derives from the perspective equation, $x_i = f(X_i/Z_i)$, where f is the focal length of the

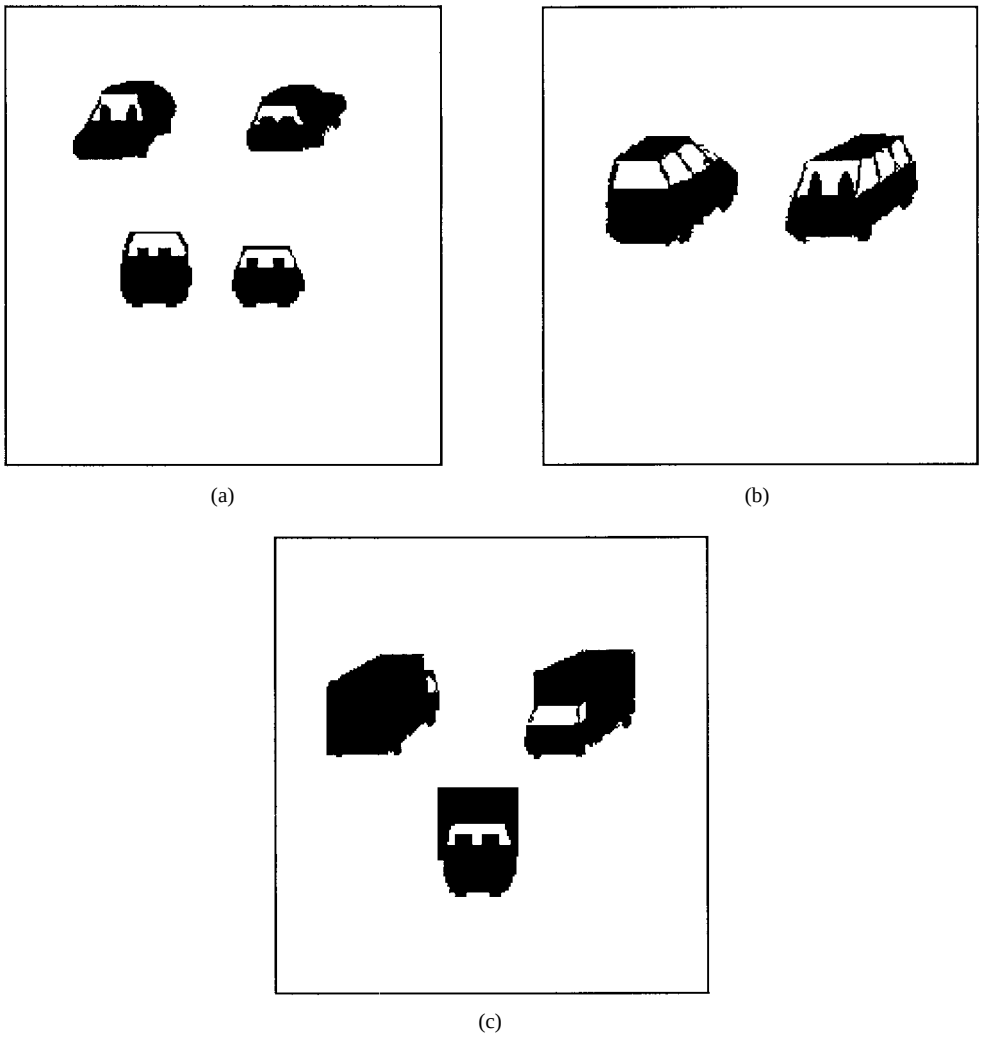


Fig. 5. Synthetic binary images representing different views of the 3-D synthetic object models of (a) a car, (b) a van, and (c) a truck.

camera, and X_i and Z_i are 3-D coordinates of the object barycenter $\boldsymbol{B_i}$ in the camera reference system. This nonlinear equation needs to be linearized around the point $(\bar{X_i}, \bar{Z_i})$ and transformed in the coordinates (X, Y, Z) of the reference system (see Appendix I)

$$x_i = \frac{f}{\overline{Z_i}} \cdot X_i + f\frac{\bar{X_i}\sin\alpha}{\overline{Z_i}^2} \cdot \frac{\lambda_i}{2} - \frac{\bar{X_i}\cos\alpha}{\overline{Z_i}^2} \cdot Z_i - f\frac{\bar{X_i}}{\overline{Z_i}^2} h_c \sin\alpha + f\frac{\bar{X_i}}{\overline{Z_i}} \tag{11a}$$

where $\overline{Z_i} = \bar{Y_i}\sin\alpha + \bar{Z_i}\cos\alpha + h_c\sin\alpha$, h_c is the height of the camera from the ground plane and λ_i is the height of the object on the image plane. Analogously, for the x_{i-1} coordinate

$$x_{i-1} = \frac{f}{\overline{Z_{i-1}}} \cdot X_{i-1} + f\frac{\bar{X}_{i-1}\sin\alpha}{\overline{Z_{i-1}}^2} \cdot \frac{\lambda_i}{2} - f\frac{\bar{X}_{i-1}\cos\alpha}{\overline{Z_{i-1}}^2} \cdot Z_{i-1} - f\frac{\bar{X}_{i-1}}{\overline{Z_{i-1}}^2} h_c \sin\alpha + f\frac{\bar{X}_{i-1}}{\overline{Z_{i-1}}} \tag{11b}$$

where $\overline{Z_{i-1}} = \bar{Y}_{i-1}\sin\alpha + \bar{Z}_{i-1}\cos\alpha + h_c\sin\alpha$.

The computation of the measurable quantity Δx_i can be found in the Appendix I.

The other QM's are represented by the 3-D object orientation θ_i, the depth Z_i of the object barycenter $\boldsymbol{B_i}$, the quantities Z_{i-1} and $\dot{Z}_i$, and the height of the projection of the object shape on the image plane λ_i. As θ_i, Z_i, Z_{i-1}, and $\dot{Z}_i$ are elements of the status vector $\boldsymbol{\Phi_i}$, there is a direct relation among these measures and the QI's. If perspective distortions are limited and the object dimensions are small with respect to the object distance from the camera, it is possible to approximate the 3-D position of the object barycenter with the 2-D position of the MBR on the image plane. Moreover, the value of the skeleton function computed in the point (x_b, y_b) can be considered approximately the 2-D height of the object shape in correspondence of the projection of the object barycenter onto the image plane [Fig. 6(c)].

C. Dynamic Model Formulation

The definition of a dynamic model for the QI's consists in the formulation of an equation which describes their temporal behavior (*dynamic system*) and of an equation which describes

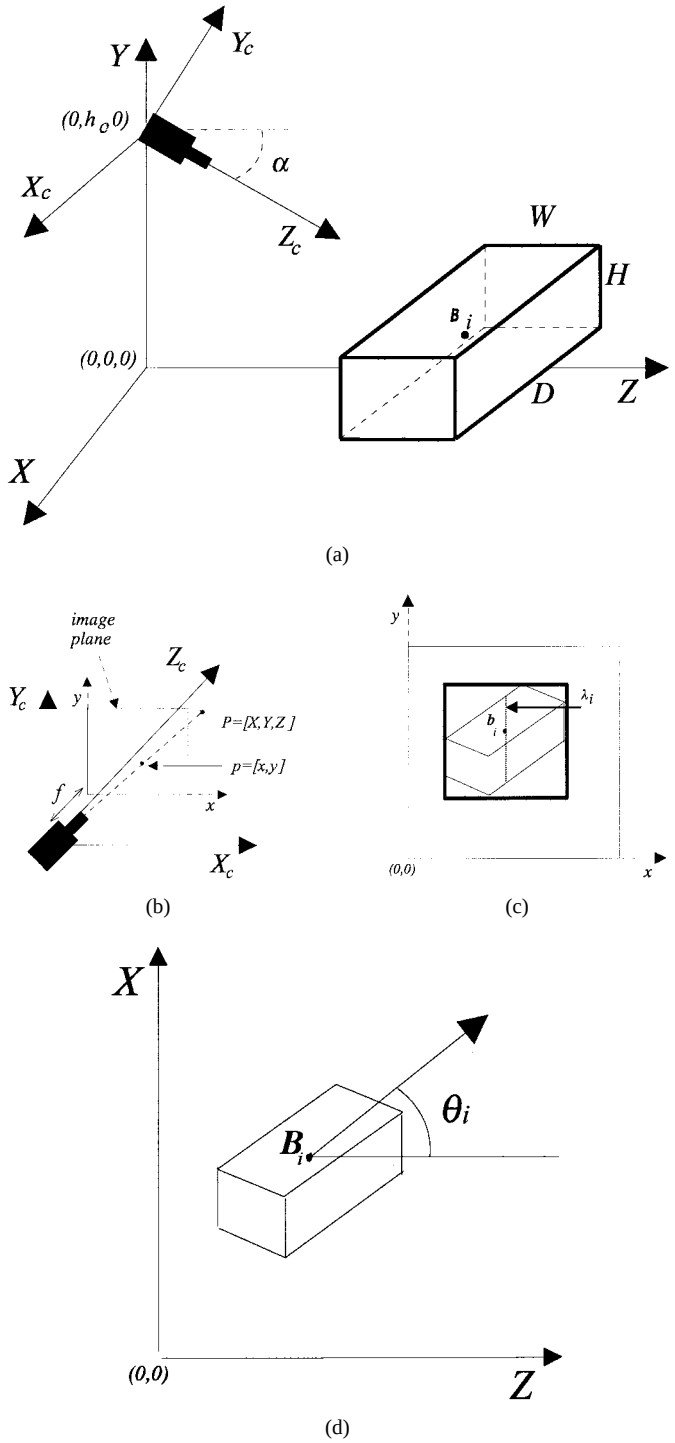


Fig. 6. (a) 3-D general reference system (X, Y, Z), (b) camera reference system (X_c, Y_c, Z_c) with tilt angle α, (c) image reference system (x, y), and (d) 3-D main object orientation.

the relation among the QI's and the QM's (*measure system*)

$$\boldsymbol{\Phi}_{i+1} = \boldsymbol{A}\boldsymbol{\Phi}_i + \xi_i \tag{12a}$$

$$\boldsymbol{y}_i = h(\boldsymbol{\Phi}_i^-) + \boldsymbol{H}_i(\boldsymbol{\Phi}_i^-)\boldsymbol{\Phi}_i^- + \boldsymbol{\eta}_i \tag{12b}$$

where $\boldsymbol{A}$ is a square matrix relating $\boldsymbol{\Phi_i}$ and $\boldsymbol{\Phi_{i+1}}$ in the absence of a forcing function, and ξ_i represents a Gaussian noise with zero mean and covariance matrix $\boldsymbol{Q}_i$. In particular, as the status vector $\boldsymbol{\Phi_i}$ is composed by two independent sets of variables, i.e., $\boldsymbol{\Phi}_{1_i} = [X_{i-1}, X_i, \dot{X}_{i-1}, Z_{i-1}, \dot{Z}_i, Z_{i-1}, \dot{Z}_i]$ and $\boldsymbol{\Phi}_{2_i} = [\theta_i, H, W, D]$, the dynamic system can be divided into two independent systems which can be considered separately. Equation (12a) can be rewritten in a matrix form

$$\begin{bmatrix}\Phi_1\\ \Phi_2\end{bmatrix}_{i+1} = \begin{bmatrix}A_1 & 0\\ 0 & A_2\end{bmatrix}\begin{pmatrix}\Phi_1\\ \Phi_2\end{pmatrix}_i + \begin{pmatrix}\xi_1\\ \xi_2\end{pmatrix}_i. \tag{13}$$

D. QI's Estimation by an Extended Kalman Filter Model

The problem to solve consists in estimating the time-varying QI's, i.e., the status vector $\boldsymbol{\Phi_i}$, by measuring the QM's only. To this aim, an EKF has been used: given the initial values, the QI's can be determined univocally by the system model [26]. Such a filter generates at each time instant an updated estimation of the state vector by means of the following two phases:

updating

$$K_i = \boldsymbol{\Sigma}_i^- \boldsymbol{H}_i^T[\boldsymbol{H}_i\boldsymbol{\Sigma}_i^-\boldsymbol{H}_i^T + \boldsymbol{R}_i]^{-1} \tag{14a}$$

$$\boldsymbol{\Phi}_i^+ = \boldsymbol{\Phi}_i^- + K_i(\boldsymbol{y}_i - h(\boldsymbol{\Phi}_i^-) - \boldsymbol{H}_i\boldsymbol{\Phi}_i^-) \tag{14b}$$

$$\boldsymbol{\Sigma}_i^+ = (\boldsymbol{I} - K_i\boldsymbol{H})\boldsymbol{\Sigma}_i^- \tag{14c}$$

where $\boldsymbol{H}_i = \boldsymbol{H}_i(\boldsymbol{\Phi}_i^-)$, K_i is the filter gain and $\boldsymbol{\Sigma}_i$ is the covariance matrix of the status vector

prediction

$$\boldsymbol{\Phi}_{i+1}^- = \boldsymbol{A}_i\boldsymbol{\Phi}_i^+ \tag{15a}$$

$$\boldsymbol{\Sigma}_{i+1}^- = \boldsymbol{A}_i\boldsymbol{\Sigma}_i^+\boldsymbol{A}_i^T + \boldsymbol{Q}_i. \tag{15b}$$

Before applying the EKF, some parameters must be initialized: (a) the covariance matrixes $\boldsymbol{R}_i$ and $\boldsymbol{Q}_i$, related to the noise affecting the dynamic and measure systems, respectively, (b) the status vector $\boldsymbol{\Phi}_0$, and (c) the covariance matrix $\boldsymbol{\Sigma}_0$. The status vector $\boldsymbol{\Phi}_0$ is initialized on the basis of some measures given on the first frames of the sequence. At the first frame, i.e., $i = 0$, the following QI's are measured: x_0 and Z_0 from which it is possible to obtain $X_0 = f \times x_0 \times Z_0$. Then, at the second and the third frame, the QI's, x_1, Z_1, and θ_1, and x_2, Z_2, and θ_2, are measured. The EKF starts at the third frame $(i = 2)$ with the following status vector:

$$\Phi_2^- = \left[X_1, X_2, \frac{X_1 - X_0}{\Delta t}, Z_1, Z_2, \frac{Z_1 - Z_0}{\Delta t}, \frac{Z_2 - Z_1}{\Delta t}\right.$$
$$\left.\theta_2, H, W, D\right].$$

VI. Results

Two real image sequences (labeled with t and c, respectively) representing two vehicles, i.e., a truck moving in an indoor parking area and a car moving in an outdoor environment, are used as test data [Fig. 7(a) and (b)]. Both vehicles are moving in a such way to cover a curve of 180° degrees. A third image sequence (labeled with m) showing the same scenario of the image sequence in Fig. 7(b), but with the car moving at a lower distance from the camera, has also

been considered. Results are presented to demonstrate at first the performances of each system module and then to analyze the behavior of the whole system. Comparisons with other existing methods for object tracking and object recognition are also given.

A. Tracking Results

In order to test the performances of the tracking module, the number of object classes is set to one ($H = 1$) and the class of the object moving in the scene is fixed (i.e., a truck for the sequence t and a car for the sequence c). In this way, the recognition module is inhibited and a single EKF is applied. The first experiment is focused on 3-D object orientation estimation on both image sequences t and c, where each object appears with $K = 18$ different orientations (10°, ..., 180°), assuming an error included in the range [−5°, +5°]. The coefficient α_s is computed by minimizing the cost function $C(\alpha_s)$ on a training sequence composed by 256 images representing a car or a truck moving in four different scenes and observed from $K = 36$ different viewpoints. The graphs in Fig. 8(a) and (b) shows, respectively, the behavior of the cost function $C(\alpha_s)$ for the image sequences t and c, where the following minimum values have been obtained: $\alpha_s = 0.43$ and $\alpha_s = 0.56$. The graphs in Fig. 8(c) and (d) show, respectively, the behaviors of the function J_i for the frames $t14$ and $c2$ of the considered image sequences. The bimodal behavior of the graphs is due to the fact that the skeleton functions extracted from images related to some complementary orientations, e.g., k and $(180° - k)$, turn out to be very similar. This is true for several man made objects which are characterized by similar views, e.g., the lateral views, and the frontal or rear views of a car. Tracking results have been evaluated on the basis of the *average least square error* σ_{ls}^2, i.e.,

$$\sigma_{ls}^2 = \frac{1}{I}\sum_{i=1}^{I} |\hat{\theta}_i - \theta_i^*|^2 \tag{16}$$

where θ_i^* is the 3-D real object orientation, $\hat{\theta}_i$ is the 3-D estimated object orientation, and I is the number of frames of the test sequence. The values $\sigma_{ls}^2 = 16.37$, $\sigma_{ls}^2 = 23.62$, and $\sigma_{ls}^2 = 26.81$ have been obtained for the sequences t, m, and c, respectively. The best result has been obtained on the image sequence t due mainly to the limited amount of noise. The illumination of the environment is controlled and constant, and the distance between the camera and the object is low (it belongs to the range [5, 25] computed in meters). To this purpose, the object surfaces are more uniform and shadows are almost inexistent. The graph in Fig. 8(e) shows the σ_{ls}^2 error versus different illumination conditions of the surveilled scene. Test images have been collected by considering the same scenario represented by the image sequence c at different hours during a spring day (from 8 a.m. to 8 p.m.). The behavior of the graph demonstrates the capability of the proposed system to work well also in presence of low illumination conditions (from 6 p.m. to 8 p.m., the σ_{ls}^2 error belongs to the range [5.4°, 6.5°]).

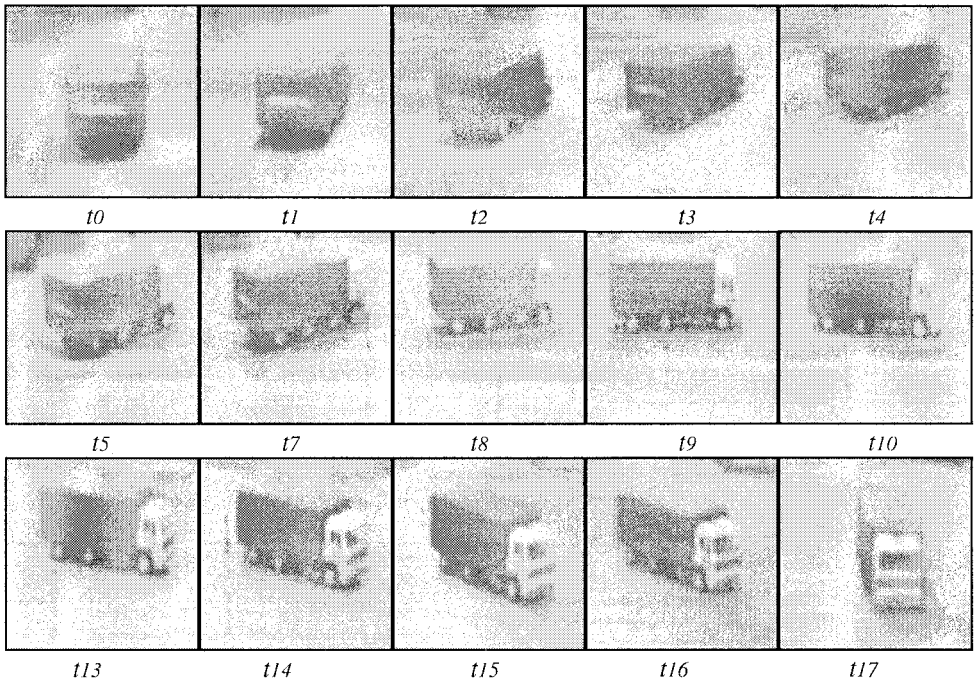
The second test on the tracking module has been performed to estimate the object trajectory on the plane XZ, i.e., the ground plane. In order to initialize the parameters of the EKF, i.e., the covariance matrices $\boldsymbol{\Sigma_2}$, $\boldsymbol{R_0}$, and $\boldsymbol{Q_0}$, several tests have been performed on several image sequences representing moving objects in real scenes. All covariance matrices are diagonal, i.e.,

$$\begin{aligned}
\mathrm{diag}(\boldsymbol{\Sigma_2}) &= [\sigma_{X_{i-1}}, \sigma_{X_i}, \sigma_{Z_{i-1}}, \sigma_{Z_i}, \sigma_{\dot{X}_{i-1}}, \sigma - \dot{X}_i, \sigma_{\dot{Z}_i} \\
&\qquad \sigma_\theta, \sigma_H, \sigma_W, \sigma_D] \\
\mathrm{diag}(\boldsymbol{R_0}) &= [\sigma_{b_i}, \sigma_{x_{i-1}}, \sigma_{\Delta x_i}, \sigma_{\theta_i}, \sigma_{Z_i}, \sigma_{Z_{i-1}}, \sigma_{\dot{Z}_i}, \sigma_{\lambda_i}] \text{ and} \\
\mathrm{diag}(\boldsymbol{Q_0}) &= [\sigma_{X_{i-1}}, \sigma_{X_i}, \sigma_{Z_{i-1}}, \sigma_{Z_i}, \sigma_{\dot{X}_i}, \sigma_{\dot{Z}_{i-1}}, \sigma_{\dot{Z}_i} \\
&\qquad \sigma_\theta, \sigma_H, \sigma_W, \sigma_D].
\end{aligned}$$

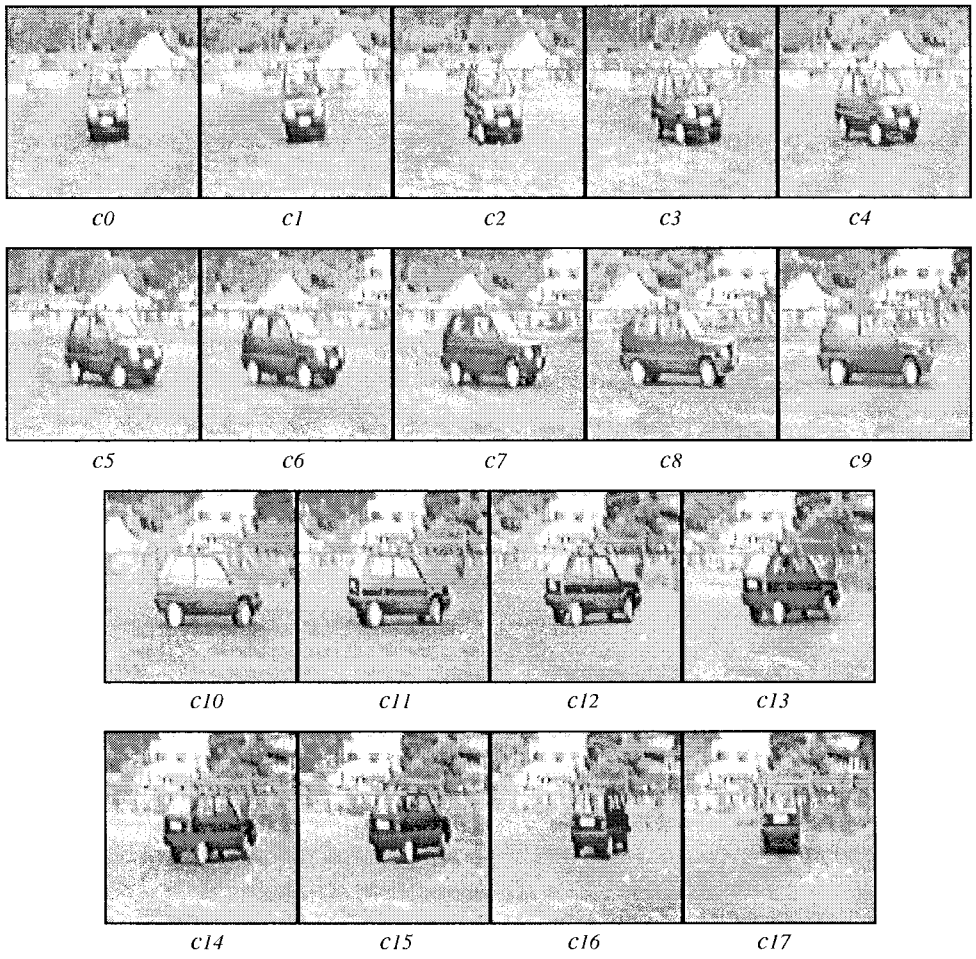
The matrix $\boldsymbol{\Sigma_2}$ is initialized by supposing (a) an initial error on object position of about to 0.5 m along both X and Z axes, (b) an initial error on object speed of about 4 m/s, (c) an initial error on object orientation included in the range [−5°, +5°], and (d) a low initial error about the object dimensions (computed by means of the MBR sizes, the camera parameters, and the camera calibration matrix), i.e., $\mathrm{diag}(\boldsymbol{\Sigma_2})$ = [0.25, 0.25, 0.25, 0.25, 2, 2, 2, 2, 100, 0.05, 0.05, 0.05]. The matrix $\boldsymbol{R_0}$ which characterizes the noise of the measure system has been initialized as follow: $\mathrm{diag}(\boldsymbol{R_2})$ = [5 × 10^{-4}, 10^{-3}, 10^{-3}, 100, 0.25, 0.25, 2, 10^{-3}]. Finally, the matrix $\boldsymbol{Q_0}$ which characterized the noise of the dynamic system has been initialized by setting to 0.25 the variance of the noise affecting the eight state variables which describe the object trajectory, to 100 the variance of the object orientation, and to 0.01 the variance of the object dimensions, i.e., $\mathrm{diag}(\boldsymbol{Q_0})$ = [0.25, 0.25, 0.25, 0.25, 0.25, 0.25, 0.25, 0.25, 100, 0.01, 0.01, 0.01]. Fig. 9(a) and (b) shows the estimated depth Z of the center of mass of the object for the sequences t and c, respectively. The real Z values, reported in the graphs, have been measured manually by an human operator. The behavior of the curve representing the estimated depth parameter is very close to the real one in both the graphs. Small values have been obtained for the σ_{ls}^2 error, i.e., 0.73 and 1.53 for the image sequences t and c, respectively, which correspond to depth estimation error of about 0, 5 and 0, 7 m. Fig. 10(a) and (b) show the object trajectory estimated by the EKF for the image sequences t and c, respectively. It is worth noting that also if an accurate initialization of the filter is not given, the estimated trajectories follow the real ones. This results are particularly important for the images of the sequence c where, despite of a complex outdoor scene, few steps are required to obtain the filter convergence.

B. Recognition Results

The performances of the recognition module have been tested by considering five different object classes ($H = 5$) (i.e., cars, motorcycles, vans, lorries and buses) taken from a limited number of viewpoints $K = 4$, i.e., 0°, 90°, 180°, 270°. Object recognition is determined through a match between the vector $\boldsymbol{V}_{si}$ related to the ith object in the analyzed image



(a)



(b)

Fig. 7. Real image sequences (labeled with (a) t and (b) c representing two vehicles, i.e., a truck moving in an indoor area and a car moving in an outdoor environment.

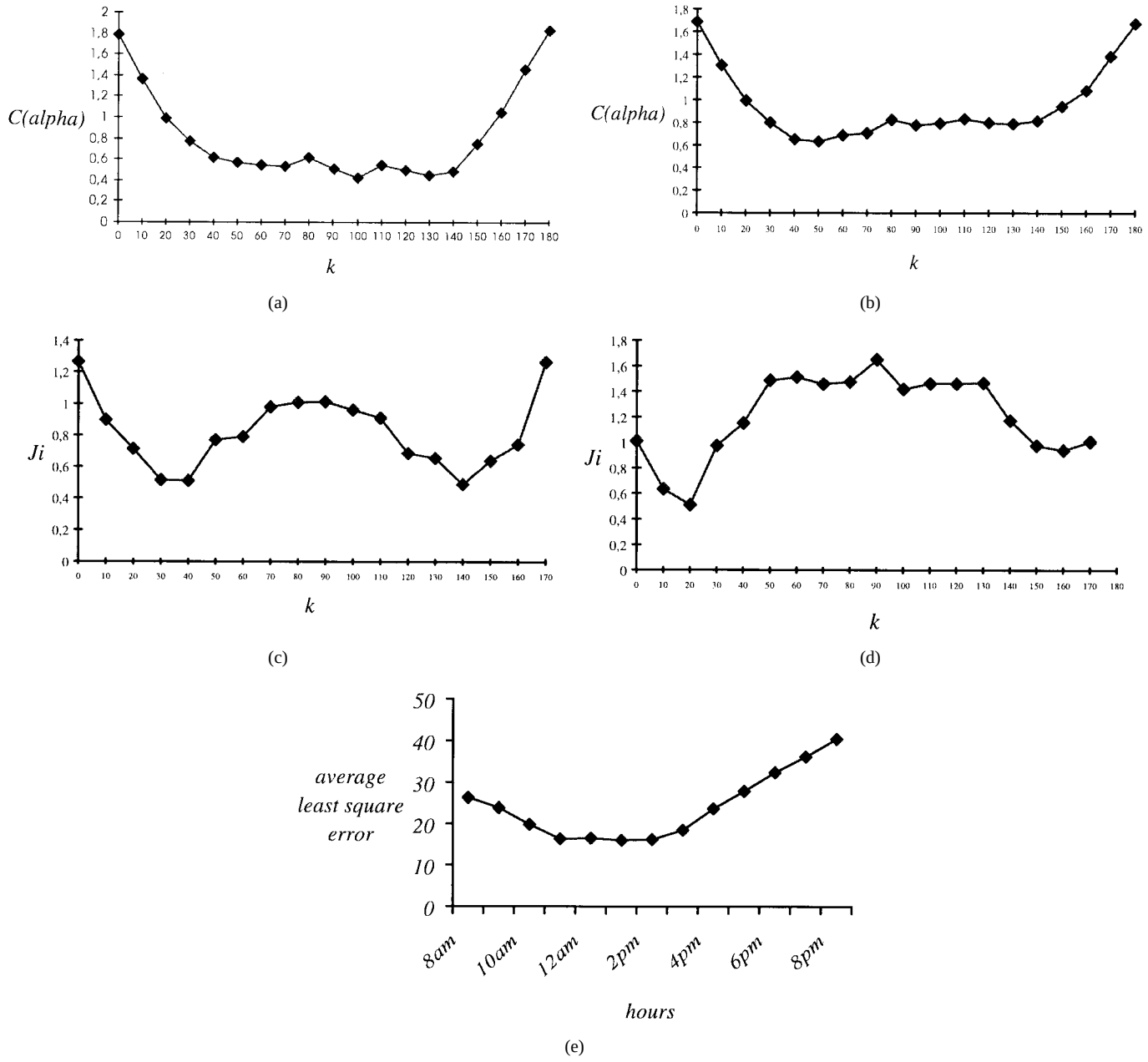
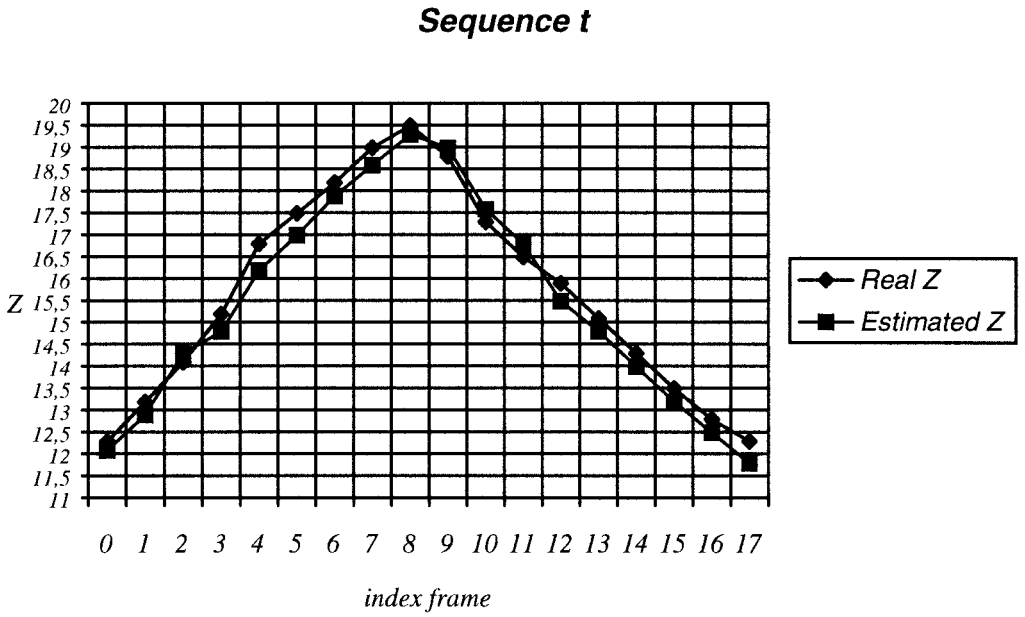


Fig. 8. Graphs representing the behavior of the cost function $C(\alpha)s)$ versus different k values for the image sequences (a) t and (b) c, graphs representing the behavior of the classification function J_i versus different k values for the frames (c) $t14$ and (d) $c2$, and (e) graph of the σ_{ls}^2 error versus different illumination conditions of the surveilled scene.

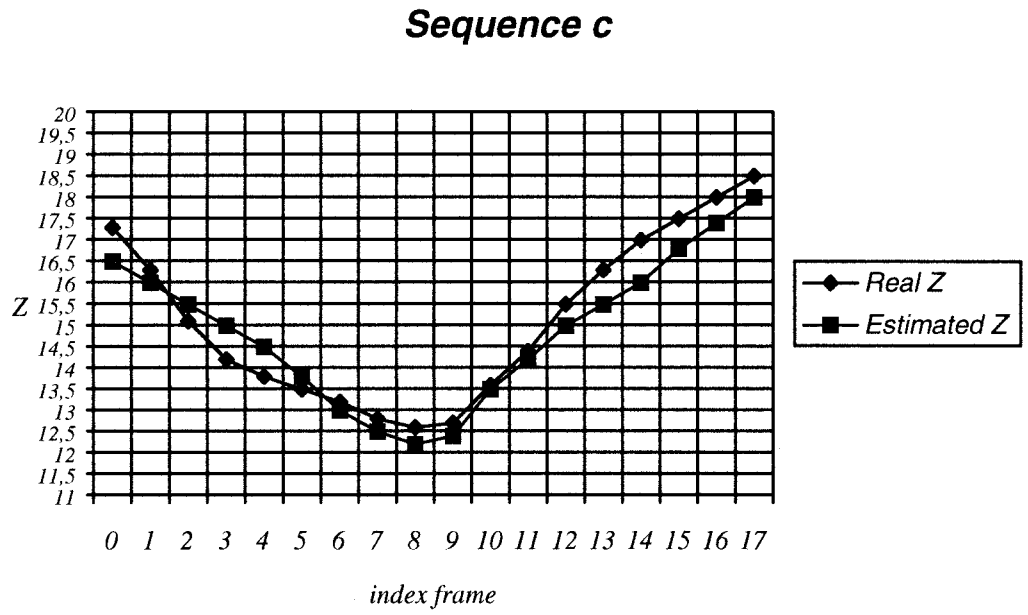
and all vector related to the known object models stored into the model database. A training sequence composed by 64 images has been used to calculate the value of the coefficient α_s., i.e., $\alpha_s = 0.65$. By fixing to three the number of consecutive frames which are necessary to recognize the object as belonging to the same class to recognize effectively the object itself, the obtained percentage of correct recognition is about 95% (the truck is correctly recognized in 17 frames on 18) and 84% (the car is correctly recognized in 15 frames on 18) for the image sequences t and c, respectively. Recognition errors are due mainly to two different causes. 1) There are some synthetic images related to different object classes whose approximated skeleton functions are quite similar, e.g., the model of the car with orientation 0° and the model of the truck with orientation 90°. 2) The noise which affects the measure of the object orientation can reduce the discrimination power of the function J_i.

C. System Performances

The performances of the proposed system have been evaluated in terms of processing times, percentage of correct object detection and recognition, and tracking accuracy on a large test set (composed by about 3×10^3 images) acquired with different background and illumination conditions. A model database composed by $H = 5$ object classes (i.e., cars, motorcycles, vans, lorries, buses), each one taken from $K = 36$ different viewpoints has been considered.



(a)

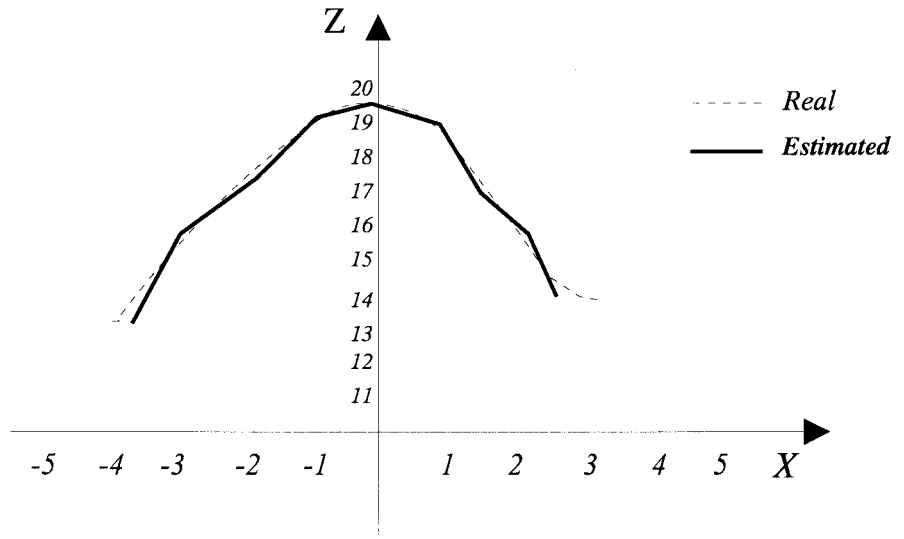


(b)

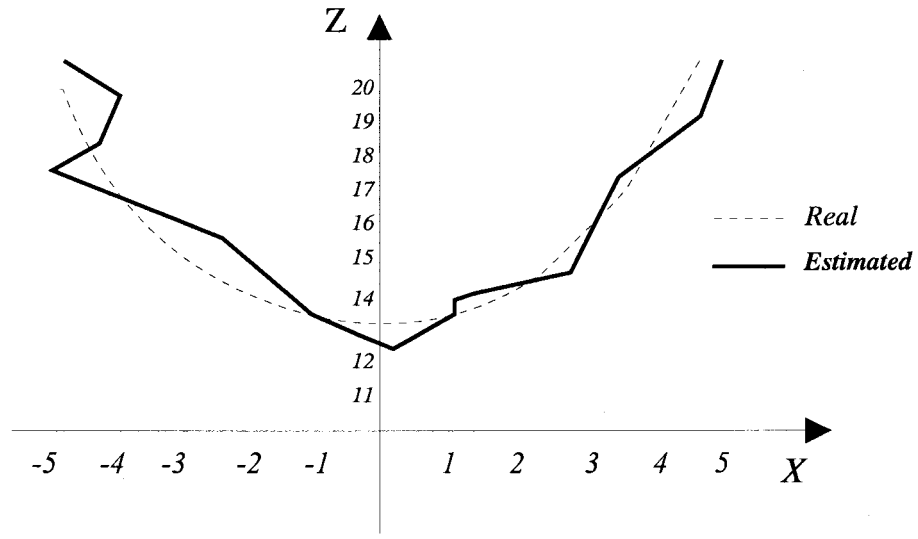
Fig. 9. Graphs of the real and estimated object depth Z computed on the sequences (a) t and (b) c.

Time performances of the system have been evaluated on a PC Pentium-II at 300 MHz. The processing of the sequences t and c (composed by 18 frames) requires about 0.32 and 0.47 s per frame, respectively. Less computation time is required to process the first sequence as it contains less noise which reduces the speed of the SMS extraction process. By testing the proposed system on the whole test set, an average frame rate of about three frames per s has been obtained. It is important to note that the processing time is mostly required by the feature extraction process and the matching operations between data and models (about 75%). The image acquisition module requires about 10% of the whole processing time, while the tracking module, the CD, and the FA modules require about 7%, 5%, and 3%, respectively.

Several experiments on the whole test set have demonstrated that the system reaches a percentage of correct object detection of about 95% and a correct object recognition of about 83% (limited to the object models contained into the database). Table I gives the percentage of correct object recognition, the distribution in other classes of bad recognition, and information about false alarms which occur when the system detects MBR's containing object shapes which cannot be assigned to any class. This is due mainly to the presence of shadows, high noise level, or combination of partial overlapping of



(a)



(b)

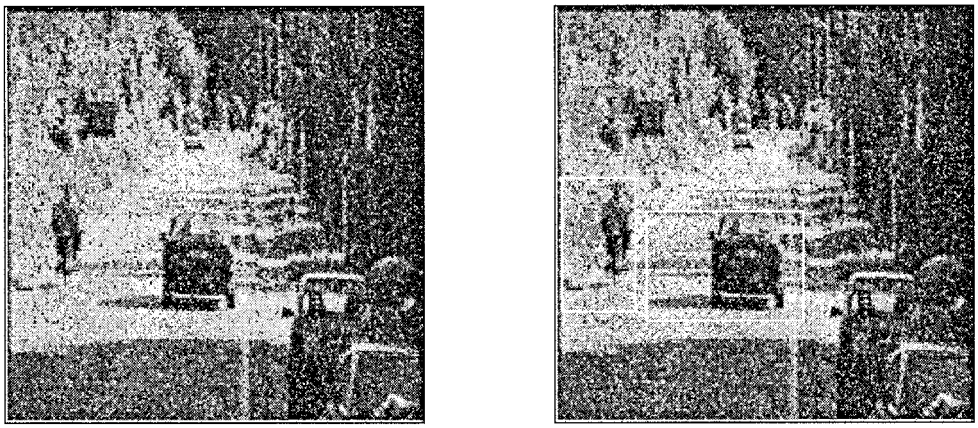
Fig. 10. Graphs of the real (*dot line*) and estimated (*continuous line*) object trajectory computed by the EKF on the sequences (a) t and (b) c.

TABLE I
PERCENTAGE OF CORRECT RECOGNITION, DISTRIBUTION IN OTHER CLASSES OF BAD OBJECT RECOGNITION AND INFORMATION ABOUT FALSE ALARMS OBTAINED ON A TEST SET COMPOSED BY ABOUT 3×10^3 IMAGES ACQUITED WITH DIFFERENT BACKGROUND AND DIFFERENT ILLUMINATION CONDITIONS

CLASS	CAR	BUS	MOTORC.	VAN	LORRY	FA
CAR	**74**	3	5	7	2	9
BUS	3	**86**	1	3	7	0
MOTORC.	6	2	**71**	4	2	15
VAN	6	2	3	**79**	5	5
LORRY	2	5	1	4	**88**	0

multiple objects and wrong estimates of the EKF. Noisy random points with uniform distribution have been added (with different percentages from 10–40%) to the original images to simulate scenes with bad environmental conditions (heavy rain). Fig. 11(a) shows a real road image containing multiple vehicles [see Fig. 2(b) for the background image] corrupted by a high level noise (about 50%): the scene understanding becomes complex also for a human operator. Fig. 11(b) shows the output image obtained by the CD module. The car and the motorcycle near to the camera have been detected, even though the related MBR's are greater. Table II gives the percentage of correct object recognition versus the increasing percentage of the noise. It is worth noting that for noise level lower that 20% the percentage of correct recognition remains acceptable (more that 65%) for all object classes except than that of motorcycles whose dimensions are too small.

The accuracy of the tracking module has been evaluated in terms of the *average least square error* σ_{ls}^2 computed by comparing the real and the estimated object position and trajectory. On the 90% of the considered images, σ_{ls}^2 values lower than 0, 22, 3, 88, and 1, 78 have been obtained for the coordinate X, the coordinate Z, and the object orientation θ_i, respectively.



(a) (b)

Fig. 11. (a) Real road image containing multiple vehicles corrupted by a high level noise (about 50%), and (b) output image obtained by the FA module.

TABLE II
PERCENTAGE OF GOOD CLASSIFICATION VERSUS INCREASING PERCENTAGE OF NOISE FOR THE SAME TEST SET USED IN TABLE I

	NOISE (%)					
CLASS	0%	5%	10%	20%	30%	40%
CAR	74	71	69	65	58	45
BUS	86	83	80	74	63	57
MOTORCYCLE	71	69	65	59	46	31
VAN	79	77	73	69	57	49
LORRY	88	86	83	75	64	59

Moreover, in presence of partially object occlusions (about 30 images), the tracking module has correctly estimated the object trajectory in the 70% of cases.

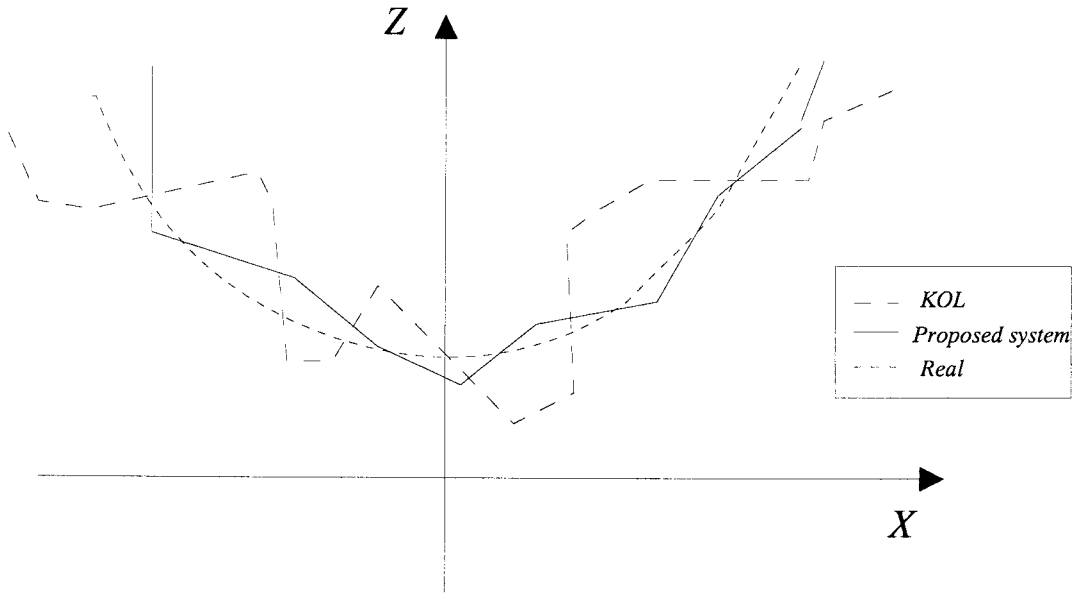
D. Comparison with Other Existing Methods

In order to demonstrate the efficiency of the proposed system, a comparison with existing model-based object tracking systems [13], [14] has been made in terms of tracking accuracy and recognition performances. The graphs in Fig. 12 show the behavior of the estimated object trajectory obtained on the image sequence c by the proposed system, the system developed by Koller *et al.* [13] (KOL), and that proposed by Malik *et al.* [14] (MAL). It is worth noting that the estimated trajectory computed by the proposed system is more accurate than these obtained by the other two systems. Table III summarizes the $\sigma^2_{\hat{l}_s}$ values obtained for the coordinate X, the coordinate Z, and the object orientation θ_i by the three compared systems. Recognition results have been compared on a set of ten real image sequences (about 100 frames) containing two different classes of objects, i.e., cars and buses. About 25% of the considered frames contain also partially object occlusions. Table IV summarizes the percentages of correct classification obtained by the considered methods.

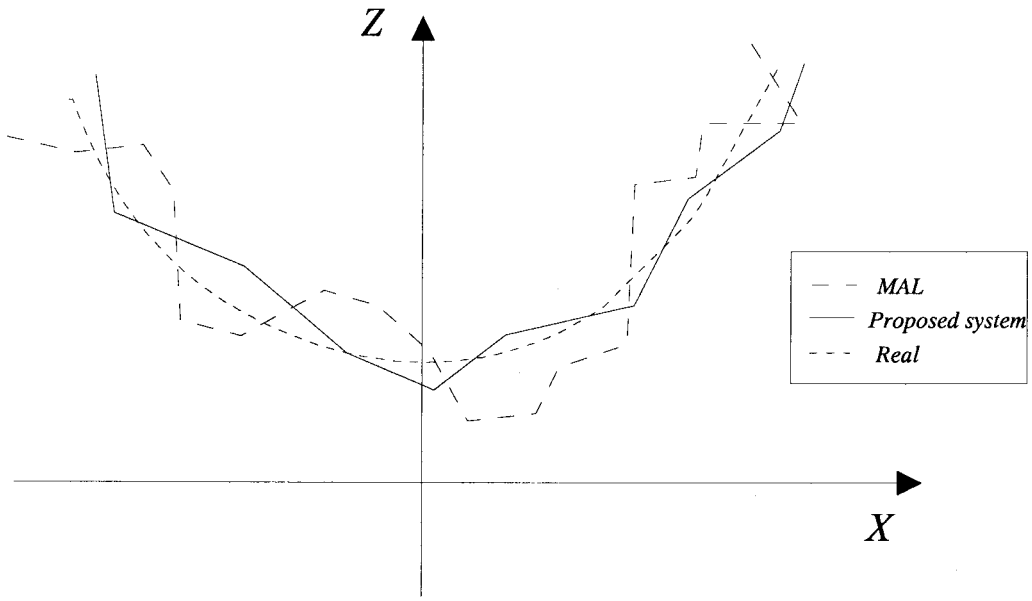
Finally, the recognition performances of the proposed system have been compared with these obtained by the method proposed by Dougherty and Cheng [31]. They proposed a morphological pattern-spectrum generated by linear structuring elements for shape recognition in the presence of edge noise. A test set composed by some binary images representing different vehicles in different positions has been considered. A noise consisting of some pixels of the shape (near the edge) being turned black and some pixels outside the shape (near the edge) being turned white has been added to the original images. Fig. 13 shows the percentage of correct object recognition obtained on the test set versus different noise densities d, i.e., density measures the number of pixels which have been modified. It is worth noting that thanks to the greater noise robustness of the SMS with respect to the morphological pattern spectrum, the proposed approach obtains a percentage of good classification greater than 65% for noise densities lower than 30%.

VII. CONCLUSIONS

A new approach for achieving both object recognition and tracking in the context of visual-based surveillance systems for outdoor environments has been presented. The core of the system consists of (a) a recognition module which performs a fast detection of unknown objects (e.g., trucks, cars, etc.) moving in the observed scene and computes an estimate of their 3-D orientation, and (b) a tracking module which predicts the position and trajectory of the detected objects. The main novelty of the proposed system is the use for both object recognition and tracking of a common feature, i.e., the statistical morphological skeleton, which achieves low computational complexity, accuracy of localization, and noise robustness. Experimental results have been focused on real scenes acquired from a static viewpoint. The performances of each system module have been evaluated in terms of processing times, percentage of correct object detection and recognition, and tracking accuracy. It was demonstrated on a large test set of real images that the proposed system is able to process about three frames per second by reaching an average correct object detection of about 95% and an average correct object recognition of about 83%. The accuracy of the tracking module has been tested on outdoor scenes containing vehicles moving at a distance from the camera ranging between 5 and 50 m. An average error of about 0.5 m along the X coordinate, 2 m along the Z coordinate, and ten degrees on the object orientation must be noticed. Comparisons with other



(a)



(b)

Fig. 12. Graphs comparing the behavior of the estimated object trajectory obtained on the image sequence c by (a) the proposed system and the KOL system and by (b) the proposed system and the MAL system. The real object trajectory is represented by the *dashed-dot line*.

TABLE III
σ_{ls}^2 VALUES OBTAINED FOR THE COORDINATE X, THE COORDINATE Z, AND THE OBJECT ORIENTATION θ_i BY THE PROPOSED SYSTEM, THE KOL SYSTEM, AND THE MAL SYSTEM

σ_{ls}^2	*Proposed system*	*KOL system*	*MAL system*
X	0.22	3.88	1.78
Z	1.23	5.71	2.23
θ_i	1.42	6.93	2.76

existing methods demonstrate the superiority of the proposed approach in both object recognition and tracking tasks. Future works will be oriented to use color cameras and to extend the functionalities of the proposed system in different application domains.

APPENDIX I

A. Linearization Process of the Measure $x_i = f(X_i/Z_i)$ Around the Point (X_i^-, Z_i^-)

The linearization process of this measure is equivalent to write the measure x_i as

$$x_i \cong g(X_i^-, Z_i^-) + \nabla g(X_i^-, Z_i^-) \cdot \begin{pmatrix} \Delta X \\ \Delta Z \end{pmatrix} \qquad \text{(a1)}$$

TABLE IV
PERCENTAGES OF CORRECT CLASSIFICATION OBTAINED BY THE THREE COMPARED METHODS ON A SET OF ABOUT 100 FRAMESCONTAINING TWO DIFFERENT CLASSES OF OBJECTS, I.E., CARS AND BUSES

	CORRECT RECOGNITION (%)		
OBJECT CLASS	Proposed system	KOL system	MAL system
CAR	76	67	72
BUS	87	76	83

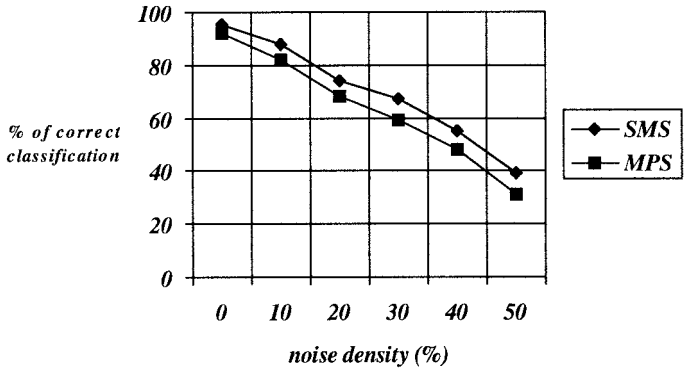


Fig. 13. Percentage of correct classification versus different noise densities d, i.e., number of modified pixels, obtained by using the statistical morphological skeleton and the morphological pattern spectrum.

where

$$g(X,Z)=f\frac{X}{Z} \quad \text{and} \quad \nabla g(X_i^-,Z_i^-)=\left(\frac{f}{Z_i^-},-f\frac{X_i^-}{Z_i^{-2}}\right).$$

By substituting equation the term ∇g in (a1), we obtain

$$\begin{aligned} x_i &\cong f\frac{X_i^-}{Z_i^-}+\frac{f}{Z_i^-}(X_i-X_i^-)-f\frac{X_i^-}{Z_i^{-2}}(Z_i-Z_i^-) \\ &=\frac{f}{Z_i^-}X_i-f\frac{X_i^-}{Z_i^{-2}}Z_i^-+f\frac{X_i^-}{Z_i^-}. \end{aligned} \tag{a2}$$

B. Linearization Process of the Measure Δx Around the Point (X_i^-,Z_i^-)

The displacement Δx can be expressed in term of the status variables as follow:

$$\begin{aligned} \Delta x_i &= f\left(\frac{X_i}{Z_i}-\frac{X_{i-1}}{Z_{i-1}}\right)=\frac{f}{Z_iZ_{i-1}}(X_iZ_{i-1}-X_{i-1}Z_i) \\ &=\frac{f}{Z_iZ_{i-1}}[(X_i-X_{i-1})(Z_{i-1}+Z_i)-X_iZ_i \\ &\quad +X_{i-1}Z_{i-1}] \\ &\cong \frac{f}{Z_iZ_{i-1}}[\dot{X}_i\Delta t(Z_{i-1}+Z_i)-X_iZ_i+X_{i-1}Z_{i-1}] \\ &=f\frac{X_{i-1}}{Z_i}f\frac{X_i}{Z_{i-1}}+\frac{f}{\dot{X}_i\Delta t(Z_{i-1}+Z_i)} \\ &=g(X_{i-1},X_i,\dot{X}_i,Z_{i-1},Z_i). \end{aligned} \tag{a3}$$

To this end, to linearize the Δx measure is needed to linearize the function g around X_i^-. It is possible to write Δx as

$$\Delta x_i=g(\cdots)+\nabla g(\cdots)|_{X_i=X_i^-}\cdot(X_i-X_i^-) \tag{a4}$$

where

$$\begin{aligned} \nabla g_{x_{i-1}} &= \frac{\partial g(\cdots)}{\partial x_{i-1}}=\frac{f}{Z_i}, \quad \nabla g_{x_i}=-\frac{f}{Z_{i-1}} \\ \nabla g_{\dot{x}_i} &= f\frac{Z_{i-1}+Z_i}{Z_iZ_{i-1}}\Delta t \\ \nabla g_{z_{i-1}} &= f\frac{\dot{X}_i}{Z_iZ_{i-1}}\Delta t-f\frac{(Z_{i-1}+Z_i)\overline{\dot{X}}_i}{Z_iZ_{i-1}^2}\Delta t \\ &\quad +f\frac{X_i}{Z_{i-1}^2} \quad \text{and} \\ \nabla g_{z_i} &= f\frac{\dot{X}_i}{Z_iZ_{i-1}}\Delta t-f\frac{(Z_{i-1}+Z_i)\dot{X}_i}{Z_i^2Z_{i-1}}\Delta t+\frac{X_{i-1}}{Z_i^2}. \end{aligned}$$

By computing the gradient of the function $g(\cdots)$ in $X_i=X_i^-$ and substituting the corresponding value in (a4), we obtain (after some simplifications)

$$\begin{aligned} \Delta x_i &= \frac{f}{Z_i^-}X_{i-1}-\frac{f}{Z_{i-1}^-}X_i+f\cdot\Delta t\frac{Z_{i-1}^-+Z_i^-}{Z_i^-Z_{i-1}^-}\dot{X}_i \\ &\quad +\beta_i\cos\alpha\cdot Z_{i-1}+\gamma_i\cos\alpha\cdot Z_i-(\beta_i+\gamma_i) \\ &\quad \cdot\frac{h_i}{2}+N_{4_i} \end{aligned} \tag{a5}$$

where

$$\begin{aligned} \beta_i &= f\cdot\Delta t\frac{\dot{X}_i^-}{Z_i^-Z_{i-1}^-}-f\cdot\Delta t\frac{(Z_{i-1}^-+Z_i^-)\dot{X}_i^-}{Z_i^-Z_{i-1}^{-2}}+f\frac{X_i^-}{Z_{i-1}^{-2}} \\ \gamma_i &= f\cdot\Delta t\frac{\dot{X}_i^-}{Z_i^-Z_{i-1}^-}-f\cdot\Delta t\frac{(Z_{i-1}^-+Z_i^-)\dot{X}_i^-}{Z_{i-1}^-Z_i^{-2}}-f\frac{X_{i-1}^-}{Z_i^{-2}} \\ N_{4_i} &= -\beta Z_{i-1}^--\gamma_iZ_i^-+h_c\sin(\beta_i+\gamma_i). \end{aligned}$$

APPENDIX II

Let us consider the first dynamic system. Two different cases should be considered according to the acquisition frequency $1/\Delta t$, where $\Delta t=(t_{k+1}-t_k)$ represents the interval between two consecutive image acquisitions, and the average object velocity v. If $1/\Delta t$ is high and v is small such that the object displacement on the image plane between two consecutive frames is limited to few pixels, the object trajectory on the ground plane XZ can be approximated by a continuous set of rectilinear segments and the following relations hold:

$$\begin{aligned} X_{i+1} &= X_i+\dot{X}_i\cdot\Delta t, \quad \dot{X}_{i+1}=\dot{X}_i+\ddot{X}_i\cdot\Delta t \\ Z_{i+1} &= Z_t+\dot{Z}_i\cdot\Delta t, \quad \dot{Z}_{i+1}=\dot{Z}_i+\ddot{Z}_i\cdot\Delta t \end{aligned} \tag{b1}$$

where

$$\ddot{X}_i=\frac{\dot{X}_i-\dot{X}_{i-1}}{\Delta t} \quad \text{and} \quad \ddot{Z}_i=\frac{\dot{Z}_i-\dot{Z}_{i-1}}{\Delta t}.$$

To this aim, the first dynamic system will be characterized by

$$A_1=\begin{bmatrix} A_{1_1} & 0 \\ 0 & A_{1_2} \end{bmatrix}, \quad \text{where}$$

$$A_{1_1}=A_{1_2}=\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 2 \end{bmatrix}$$

and by ξ_{1_i} which represents a Gaussian noise with zero mean and covariance matrix Q_{1_i}. If v is high such that the object displacement between two consecutive frames is high, the object trajectory on the ground plane XZ is approximated by Hermite splines [18]. To this purpose, at the time instant $(i+1)$, the object barycenter $\boldsymbol{B}_{i+1}(X_{i+1}, Z_{i+1})$ is constrained to belong to the Hermite spline with endpoints $\boldsymbol{B}_{i-1}(X_{i+1}, Z_{i+1})$ and $\boldsymbol{B}_i(X_i, Z_i)$ and with tangent lines to these endpoints parallel to $(\dot{X}_{i-1}, \dot{Z}_{i-1})$ and $(\dot{X}_i, \dot{Z}_i)$. The following Hermite spline is selected for the definition of the dynamic system:

$$\begin{aligned} P_0 &= B_{i-1}(X_{i-1}, Z_{i-1}), \quad P_1 = B_i(X_i, Z_i) \\ R_0 &= \dot{B}_{i-1}(\dot{X}_{i-1}, \dot{Z}_{i-1}), \quad R_1 = \dot{B}_i(\dot{X}_i, \dot{Z}_i) \end{aligned} \tag{b2}$$

and the length of the spline between two time instants t_i and t_{i-1} can be represented as

$$\begin{aligned} Q_{i-1}(t) &= [x_{i-1}(t), y_{i-1}(t), z_{i-1}(t), 1] \\ &= [(t-t_{i-1})^3, (t-t_{i-1})^2, (t-t_{i-1}), 1] \\ &\quad \cdot M_H G_{H_{i-1}} \end{aligned} \tag{b3}$$

where

$$M_H = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad \text{and}$$

$$G_{H_{i-1}} = \begin{bmatrix} P_{i-1}(x(i-1), y(i-1), z(i-1)) \\ P_1(x(i), y(i), z(i)) \\ R_0(\dot{x}(i-1), \dot{y}(i-1), \dot{z}(i-1)) \\ R_1(\dot{x}(i), \dot{y}(i), \dot{z}(i)) \end{bmatrix}.$$

The position and the speed of the object barycenter at the time instant $(i+1)$ are given by

$$\begin{aligned} B_{i+1} &= (X_{i+1} \quad 0 \quad Z_{i+1}) = Q_{i-1}(t_{i+1} = t_i + \Delta t) \\ &= (x(t_{i+1}) \quad 0 \quad z(t_{i+1})) \\ \dot{B}_{i+1} &= (\dot{X}_{i+1} \quad 0 \quad \dot{Z}_{i+1}) = \dot{Q}_{i-1}(t_{i+1} = t_i + \Delta t) \\ &= (\dot{x}(t_{i+1}) \quad 0 \quad \dot{z}(t_{i+1})) \end{aligned} \tag{b4}$$

and, consequently, we obtain [18]

$$\begin{aligned} X_{i+1} &= 5X_{i-1} - 4X_i + 2\Delta t \cdot X_{i-1} + 4\Delta t \cdot X_i \\ \dot{X}_{i+1} &= \frac{12}{\Delta t} X_{i-1} - \frac{12}{\Delta t} X_i + 5\dot{X}_{i-1} + 8\dot{X}_i. \end{aligned} \tag{b5}$$

Analogue equations can be found for the Z coordinate. Finally, the A_1 matrix for the dynamic system is obtained as

$$A_1 = \begin{bmatrix} A_{1_1} & 0 \\ 0 & A_{1_2} \end{bmatrix}, \quad \text{where}$$

$$A_{1_1} = A_{1_2} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 5 & -4 & 2\Delta t & 4\Delta t \\ 0 & 0 & 0 & 1 \\ \dfrac{12}{\Delta t} & -\dfrac{12}{\Delta t} & 5 & 8 \end{bmatrix}.$$

The second system is composed by four independent equations which describe the behavior of the following status variables: θ_i, H_i, W_i, and D_i. The behavior of the variable θ_i can be described by the following:

$$\theta_{i+1} = \theta_i + \delta_i + \xi_{9_i} \tag{b6}$$

where

$$\delta_i = \begin{cases} k & \text{if } \theta_i > \theta_{i-1} \text{ and } \theta_i \neq 180 \\ -k & \text{if } \theta_i < \theta_{i-1} \text{ and } \theta_i \neq 0 \\ 0 & \text{otherwise.} \end{cases}$$

The parameter k is *a priori* determined and depends on the acquisition frequency and on the average speed of the object; ξ_{9_i} represents the nine component of the noise vector ξ_i (it is a Gaussian variable with zero mean and variance q_{9_i}). Finally, the equations related to the object dimensions can be written as follows:

$$\begin{aligned} H_{i+1} &= H_i + \xi_{10_i} \\ W_{i+1} &= W_i + \xi_{11_i} \\ D_{i+1} &= D_i + \xi_{12_i} \end{aligned} \tag{b7}$$

where ξ_{10}, ζ_{11}, ζ_{12} represent three different Gaussian noises with zero mean and low variances $q_{10_i}, q_{11_i}, q_{12_i}$, respectively. Finally, we obtain

$$\boldsymbol{A}_2 = \boldsymbol{I} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \xi_{2_i} = \begin{pmatrix} \xi_{9_i} \\ \xi_{10_i} \\ \xi_{11_i} \\ \xi_{12_i} \end{pmatrix}.$$

REFERENCES

[1] A. F. Toal and H.Buxton, "Spatio-temporal reasoning within a traffic surveillance system," in *Proc. 2th Eur. Conf. Comput. Vision*, S. Margherita, Italy, 1992, pp. 884–892.

[2] E. F. Lyon, "The application of automatic surface lights to improve airport safety," *IEEE AES Syst. Mag.*, Mar. 1993, pp. 14–20.

[3] R. J. Howarth, "Spatial representation, reasoning and control for a surveillance system," Ph.D. dissertation, Queen Mary and Westfield College, Univ. London, U.K., 1994.

[4] D. Corrall, "VIEW: Computer vision for surveillance applications," *IEE Colloquium Active Passive Techniques for 3-D Vision*, IEE, London, U.K., vol. 8, pp. 1–3, 1991.

[5] R. A. Brooks, "Model-based 3-D interpretation of 2-D images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 5, pp. 140–150, May 1983.

[6] P. Fua and A. J. Hanson, "Using generic geometric models for intelligent shape extraction," in *Proc. DARPA Image Understanding Workshop*, Los Angeles, CA, 1987, pp. 227–233.

[7] H. G. Barrow and J. M. Tenenbaum, "MSYS: A system for reasoning about scenes," *Tech. Note 121, Artificial Intell. Cent., SRI Int*, Apr. 1976.

[8] T. M. Strat and M. A. Fischler, "Content-based vision: Recognizing objects using information from both 2-D and 3-D imagery," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, pp. 1050–1065, Oct. 1991.

[9] S. Ulman, *High-Level Vision-Object Recognition and Visual Cognition*. Cambridge, MA: MIT Press, 1996.

[10] D. B. Gennery, "Visual tracking of known 3-D objects," *Int. J. Comput. Vision*, vol. 7, no. 3, pp. 243–270, 1992.

[11] D. G. Lowe, "Robust model-based motion tracking through the integration of searching and estimation," *Int. J. Comput. Vision*, vol. 8, no. 2, pp. 113–122, 1992.

[12] F. Meyer and P. Bouthemy, "Region-based tracking using affine motion models in long image sequences," *Comput. Vision, Graphics, Image Proc.: Image Understanding*, vol. 60, no. 2, pp. 119–140, 1994.

[13] D. Koller, K. Daniilidis, and H. Nagel, "Model-based object tracking in monocular image sequences of road traffic scenes," *Int. J. Comput. Vision*, vol. 10, pp. 257–281, 1993.

[14] J. Malik, D. Koller, and J. Weber, "Robust multiple car tracking with occlusion reasoning," *Eur. Conf. Comput. Vision*, Stockolm, Sweden, 1994, pp. 189–196.

[15] G. L. Foresti, "A real-time system for video surveillance of unattended outdoor environments," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 8, pp. 697–704, 1998.

[16] P. Maragos, "Pattern spectrum and multiscale shape representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 701–716, July 1989, .
[17] G. L. Foresti, C. S. Regazzoni, and A. N. Venetsanopoulos, "Coding of noisy binary images by using statistical morphological skeleton," *IEEE Workshop Nonlinear Signal Processing*, Cyprus, Greece, 1995, pp. 354–359.
[18] G. L. Foresti and C. S. Regazzoni, "A real-time model based method for 3-D object orientation estimation in outdoor scenes," *IEEE Signal Proc. Lett.*, vol. 4, pp. 248–251, 1997.
[19] ——, "A change detection method for multiple object localization in real scenes," *IEEE Conf. Indust. Electron.*, Bologna, Italy, 1994, pp. 984–987.
[20] P. Gamba, M. Lilla, and A. Mecocci, "A fast algorithm for target shadow removal in monocular color sequences" *IEEE Conf. Image Proc.*, Lausanne, Switzerland, 1997, pp. 436–439.
[21] L. Piegl and W. Tiller, "Curve and surface construction using rational B-splines," *Computer-Aided Design*, vol. 19, pp. 485–498, 1987.
[22] H. Blum, "An associative machine for dealing with the visual field and some of its biological implications" in *Biological Prototypes and Synthetic Systems*, E. E. Bernard and M. R. Kare, Eds. New York: Plenum, 1962, pp. 244–260.
[23] J. Serra, *Image Analysis and Mathematical Morphology*. New York: Academic, 1983.
[24] P. Maragos and R. W. Schafer, "Morphological skeleton representation and coding of binary images," *IEEE Trans. Acoustic, Speech, Signal Proc.*, vol. 34, pp. 1228–1244, 1986.
[25] G. L. Foresti and C. S. Regazzoni, "Properties of binary statistical morphology," *13th Int. Conf. Pattern Recognition*, Vienna, Austria, Aug. 25–29, 1996, pp. 631–635.
[26] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution, and Bayesian restoration of images," *IEEE Trans. Pattern Anal. Machine Intell.*, no. 6, pp. 721–741, 1984.
[27] B. Zeng, M. Gabbouj, and Y. Neuvo, "A unified design method for rank order, stack, and generalized stack filters based on classical Bayes decision," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 1003–1020, Sept. 1991.
[28] A. T. Hamdy, *Operation Research*. New York: Macmillan , 1976, pp. 568–571.
[29] G. L. Foresti, P. Matteucci, and C. S. Regazzoni, "A real-time approach to 3-D object tracking in complex scenes," *Electron. Lett.*, vol. 30, no. 6, pp. 475–477, 1994.
[30] R. Y. Tsai "An efficient and accurate camera calibration technique for 3-D machine vision," in *IEEE Comp. Soc. Conf. CVPR*, Miami Beach, FL, 1986, pp. 234–238.
[31] E. R. Dougherty and Y. Chen, "Morphological pattern-spectrum classification of noisy shapes: Exterior granulometries," *Pattern Recognition*, vol. 28, no. 1, pp. 81–98, 1995.

Gian Luca Foresti (S'93–M'95) was born in Savona, Italy, in 1965. He received the laurea degree in electronic engineering in 1990 and the Ph.D. degree in computer vision and signal processing in 1994 from University of Genoa, Italy.

In 1994 he was visiting Professor at Trento University, Italy, in an electronic engineering course. Currently, he is an Assistant Professor at the Department of Computer Science (DIMI) of the University of Udine, Italy. Immediately after the laurea degree, he worked with the Department of Biophysical and Electronic Engineering (DIBE) of Genoa University in the area of Computer Vision, Image Processing, and Image Understanding. His Ph.D. thesis dealt with distributed systems for analysis and interpretation of real image sequences. His main interests involve multisensor data processing for intelligent mobile vehicles, 3-D scene understanding for surveillance and monitoring applications, pattern recognition, and neural networks. He worked at several national and international project founded by the European Commission, especially in the fields of autonomous vehicle driving and surveillance systems for outdoor environments. He is author or co-author of more than 100 papers published in international journals and conferences. He served as a reviewer for several international journals. He has been also involved as an evaluator of project proposals in some CEC programs (MAST III 95, Long Term Research 95-98, and BRITE-EURAM-CRAFT 96).

Dr. Foresti is a member of IAPR.