3D Road Scene Interpretation for Autonomous Vehicle Driving

Gian Luca Foresti[1] and Carlo Regazzoni[2]

[1] Department of Mathematics and Computer Science (DIMI), University of Udine, Italy
[2] Department of Biophysical and Electronic Engineering (DIBE), University of Genova, Italy

In this paper, the problem of 3D road scene interpretation for autonomous vehicle driving is addressed. In particular, the problems of road detection and obstacle avoidance in outdoor environments are investigated. A set of descriptive primitives (straight and circular line segments) is selected to describe 3D objects which commonly occur in road scenes, e.g., people, cars, trucks, houses, etc. First, these primitives are extracted directly from the input image of the scene, and then are grouped according to specific geometric relationships (symmetry, convergence, parallelism, closeness, etc.). Relational geometrical knowledge of the elements of a group can be used to index an object in a pure bottom-up way, so decreasing the recognition complexity by reducing the amount of data to be matched with an object model database. Results on a road image containing obstacles, which show the efficiency, accuracy and time performances of the proposed method are reported.

Keywords: Image processing, feature extraction, feature grouping, autonomous vehicle driving

1. Introduction

Detection and recognition of the navigation site and of possible obstacles is a basic task for autonomous vehicle driving [1-5]. In a road environment, this task requires that a visual recognition system be able to extract a complete description of the content of an image in order to identify the road and the objects that may be on it. Many interesting objects can be perceived by such a system as compositions of regular surfaces (e.g., cars, roads, traffic signs, etc.). In general, 2D projections of regular surfaces onto the image plane are characterized by piecewise smooth contours and connected regions. Therefore, descriptive primitives (DPs), such as edges and regions, can be used to describe projections of interesting objects onto the image plane [3,4,5]. For instance, edges characterized by constant curvatures (i.e., circular and straight lines) can provide most of the information necessary to interpret a road scene. Straight line segments can be associated with parts of the car or human body (door, roof, legs, arms, etc.), while circular line segments can be associated with car wheels or human heads.

Several methods to extract straight or circular lines from an image are provided in the literature. The most used is the Hough Transform (HT), which is an efficient technique for line and curve detection in images; it was proposed by Hough [6] and improved by Duda and Hart [7] and Ballard [8]. The main limitations of the HT are loss of spatial information in the transformation process and a high false-alarm rate due to discretization effects and to the presence of spurious peaks. The non-accidentaliness principle states that the detection of certain configurations of DPs in an image is very likely related to the presence of man made objects [9] in the observed scene. For example, sets of parallel lines in a 2D image usually correspond to parallel lines in a 3D space, as well as proximal lines in a 2D image are spatially close in 3D space. Further steps are necessary to identify and locate the related objects. In particular, if one describes an object as a set of regular surfaces, under some hypothesis, the problem can be reduced to find correspondences between the set of straight or circular lines on the image plane and the 3D boundaries of each surface. This problem has many possible solutions: hypothesis-and-test approaches that imply the search for the solution in a wide space is available, but it has been demonstrated that algorithms following these approaches have to deal with a non-polynomial complexity. However, Lowe [9] suggested an alternative approach by showing that the use of perspective-invariant rules together with perceptual organization mechanisms can drastically reduce the search complexity. On the basis of his work, many authors [10,11] proposed different approaches to the so-called "grouping" problem, which aims at identifying consistent groups of DPs by using only domain-independent and viewpoint-invariant knowledge. The main advantage of grouping is to reduce the amount of data to be matched with an object model. Relational geometrical knowledge on the elements of a group can be used to index an object in a pure bottom-up way, so decreasing the recognition complexity. In [12], a probabilistic approach to grouping based on Markov Random Fields (MRFs) is proposed, which is used in this paper to obtain the most consistent groups from a road-scene image. Such groups are then compared with complex models of objects contained in a database of the recognition system (e.g. cars, pedestrians, and roads).

This paper describes a recognition system for autonomous vehicle driving. The system integrates the capability of bottom-up grouping with the possibility of propagating expectations from the 3D world of object models down to the image plane. Particular attention is devoted to reporting grouping results. In Section 2, a general description of the system is given. Section 3 deals with the description of low level modules whose main task is to extract 2D straight and circular lines from 2D images. Section 4 presents the method to group such DPs into consistent subsets. Section 5 is focused on top-down hypothesis propagation and on the matching process between groups and object surfaces. Results on real images are provided in Section 6.

2. System Description

3D interpretation of a complex scene is a crucial problem in Computer Vision that cannot be solved by means of a single or small set of methods. Generally, it is addressed by dividing the task into several subtasks of reduced complexity [6,12,13]. In this work, a system architecture composed of four levels has been developed (Fig. 1). For the present application, 256**x**256 b/w images of an outdoor real environment are acquired by a visual sensor (e.g., a CCD camera). The first level is represented by an edge-extractor algorithm [14] (i.e., the Canny operator), which generates an edge-map of the observed scene. The second level consists of a virtual sensor which is in charge of extracting straight and circular segments from the edge-map. A voting-based approach is applied to map the edge-pixel information directly into a symbolic representation of the straight or circular segments present in the scene. The numerical input information consists of some quantities provided by the Canny algorithm: (a) the coordinates (x, y) of each edge point $\mathbf{x}$ on the image plane, and (b) the gradient orientation $\gamma(\mathbf{x})$ computed for such coordinates. Then, detected segments are grouped according to specific geometric relationships (e.g., symmetry, strong or weak convergence, parallelism, closeness, etc.) in order to reduce the amount of data to be matched with an object model database. The output is represented by a symbolic graph, called the *Descriptive Primitive Graph* (DPG), where each node is associated with either a straight or a circular segment in the image and the link between two nodes represents a different geometric relationship between segments.

At the third level, segment-grouping operations are performed according to perceptual organization rules [9]. This stage consists of assigning an additional label to each node in order to detect groups of DPs characterized by a given set of relations. To this end, a Markov Random Field (MRF) process is applied to the DPG graph to aggregate segments in order to detect a possible obstacle in the scene. For example, to detect a car (e.g., lateral view), the MRF process associates a couple of close parallel segments and two circles with the lateral surface of the car. The grouping process follows a Maximum Probability (MP) approach to estimate the best label configuration by minimizing an adequate energy function [12]. A stochastic relaxation labelling based on Simulated Annealing [16] with the Metropolis sampler is followed to obtain the best configuration. Thanks to the reduced dimensionality of the graph, as compared with pixel-based MRF approaches [15,16], computation time can be reduced to an acceptable value, even in the case of complex scenes. The number

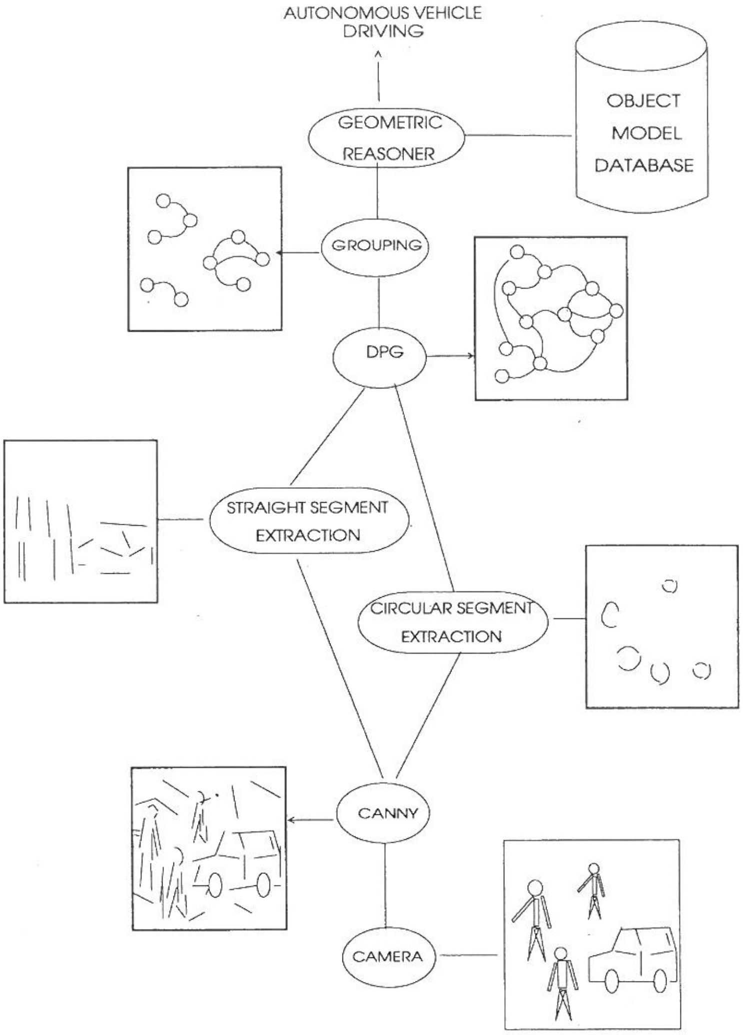

AUTONOMOUS VEHICLE DRIVING
GEOMETRIC REASONER
OBJECT MODEL DATABASE
GROUPING
DPG
STRAIGHT SEGMENT EXTRACTION
CIRCULAR SEGMENT EXTRACTION
CANNY
CAMERA

Fig. 1. General architecture of the system.

of DPs in a scene is typically at least two orders of magnitude smaller than the number of pixels, and the average connectivity between the nodes of the DPG is only a little higher than the typical 4-connectivity between image pixels. The grouping process produces, as a result, a list of groups of descriptive primitives.

The last system level performs the final scene description, i.e., a 3D reconstruction of all objects obtained by using a Geometric Reasoner module [6,13] according to the a-priori geometric knowledge of the objects and to the sensor models. To this end, top-down hypotheses about the object's poses are made by fixing hypothesized rototranslation matrices between the sensor reference system and the reference system in which the object is described.

3. Low Level Modules

Two different kinds of DPs are considered for the road scene interpretation task: (a) straight segments, which are used to approximate on the image plane the contours of vehicle surfaces (e.g., roof, door, bonnet, etc.) or the legs, arms and trunks of people, and (b) circular segments, which are used to represent the head of a person or the wheels of a vehicle.

A. DPs extraction

DPs are obtained by applying, in a sequential way, an edge extraction algorithm and a voting mechanism. The former uses, as input data, an original image $I(\mathbf{x}) = I(x, y)$ (where $\mathbf{x} = (x, y)$ is an image point), from which it extracts an edge image $E(\mathbf{x})$ and a gradient image $G(\mathbf{x})$ by

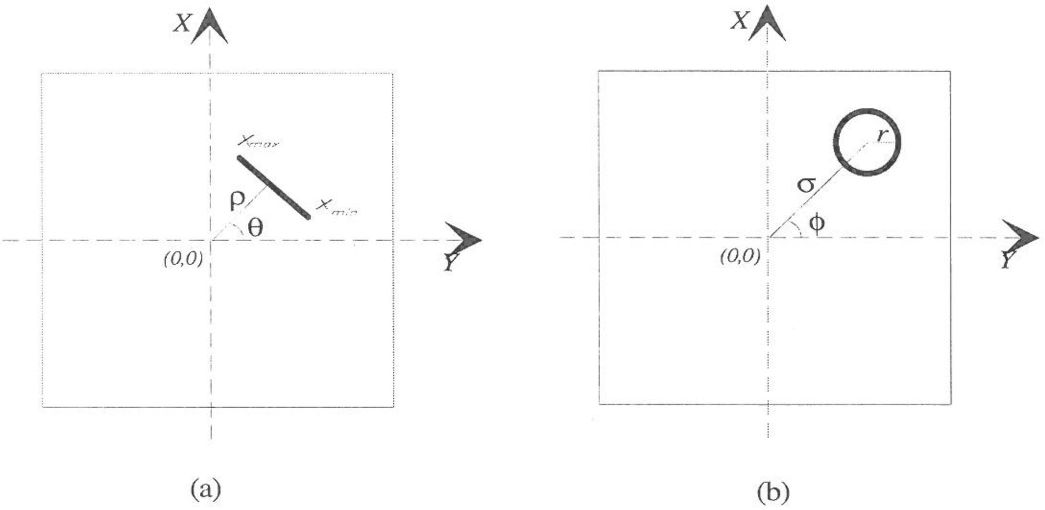


Fig. 2. 2D image representation of (a) straight and (b) circular segments.

means of a Canny filter operator [14]. These images can be defined as:

$$E(\mathbf{x}) = \{\mathbf{x} : \mathbf{x} \in I(\mathbf{x}),\ F(x) = 1\} \tag{1a}$$
$$G(\mathbf{x}) = \{\gamma(\mathbf{x}) : \mathbf{x} \in I(\mathbf{x}),\ F(x) = 1\} \tag{1b}$$

where $\gamma(\mathbf{x}) = \gamma(x, y)$ is the orientation of the gradient computed for the co-ordinates (x, y) and $\begin{cases} F(\mathbf{x}) = 1 & \text{if } \mathbf{x} \text{ is an edge point} \\ 0 & \text{otherwise} \end{cases}$. Two different voting functions are used to extract, from the edge image $E(\mathbf{x})$, the two kinds of interesting DPs: (a) straight and (b) circular segments. A straight line, $l(\pi)$, is identified on the image plane by an orientation parameter θ and by a localization parameter ρ, i.e., $\pi = (\rho, \theta)$. Each straight line can be composed by a set of collinear segments, i.e, $l(\pi) = \{l_j(\pi) : j \in [1, J(\pi)]\}$, each one univocally characterized by a vector of local parameters $[l_j, (\mathbf{x}_{\min_j}, \mathbf{x}_{\max_j}, V_j]$, where l_j is a local label, $(\mathbf{x}_{\min_j}, \mathbf{x}_{\max_j})$ are the endpoints of the j-th segment, and V_j is the number of votes associated with the j-th segment, where $J(\pi)$ represents the number of straight segments belonging to the straight line identified by π (Fig. 2a). The parameter space Π consists of an accumulator array $H(\pi) = H(\rho, \theta)$, with $2\Theta \cdot (2R + 1)$ cells ($\Theta = 180$ and $R = \frac{N}{2}\sqrt{2}$ indicate the maximum resolution of the parameters θ and ρ, respectively). For each edge point $x \in E(\mathbf{x})$, a limited set of $H(\pi)$ cells is incremented by considering the gradient information $\gamma(\mathbf{x})$ (i.e., $\gamma(\mathbf{x}) - th \leq \theta \leq \gamma(\mathbf{x}) + th$):

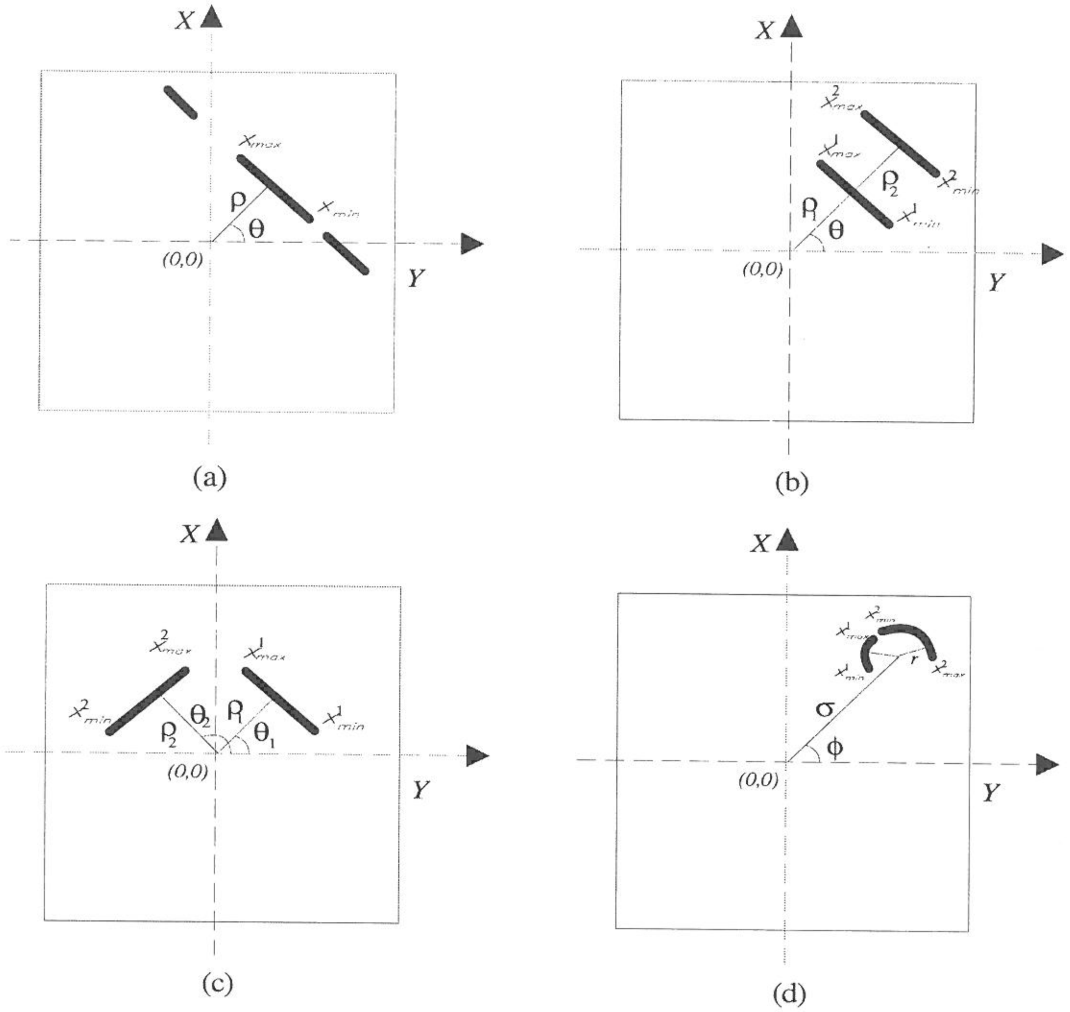
$$H(\pi) = H(\rho, \theta) = \sum_{x=1}^{N} \sum_{y=1}^{N} E(x, y) \cdot \delta[f(x, y, \rho, \theta)] \tag{2}$$

where $\delta(\cdot)$ is the normalized Kroenecker function and $f(\mathbf{x}, \pi) = f(x, y, \rho, \theta) = \rho - x \cdot cos\theta - y \cdot sin\theta = 0$ is the voting equation [6-8]. At the same time, elements of the vector L_j are updated.

Analogously, each circle $c(\varphi)$ can be univocally represented on the image plane by three parameters: an orientation parameter ϕ, a localization parameter σ and the radius r, i.e., $\varphi = (\phi, \sigma, r)$ (Fig. 2b). Each circle can be composed of a set $c_n(\varphi)$ of $N(\varphi)$ circular arcs, $c(\varphi) = \{c_n(\varphi) : n \in [1, N(\varphi)]\}$, each one characterized a vector of local parameters $[c_n, (\gamma_{\min_n}, \gamma_{\max_n}), U_n]$, where c_n is a local label, $(\gamma_{\min_n}, \gamma_{\max_n})$ are the endpoints of the n-th circular arc, and U_n is the number of votes received by the n-th circular segment. By considering the polar equation of the circle and by substituting the values of the centre (expressed in polar coordinates, i.e., $x_0 = \sigma \cdot \cos\phi$ and $y_0 = \sigma \cdot \sin\phi$), the following voting equation is obtained [17]:

$$s(x, y, \sigma, \phi, r) = \begin{cases} \sigma \cdot \cos\phi = x \pm r \cdot \cos\gamma \\ \sigma \cdot \sin\phi = y \pm r \cdot \sin\gamma \end{cases} \tag{3b}$$

Accumulator array cells (*maxima*) with a large number of votes represent straight segments (circular arcs) in the input image. A local maximum π^* is defined as a parameter-space point



(a) (b) (c) (d)

Fig. 3. (a,b,c) Collinear, parallel and convergent straight segments, and (d) collinear circular segments.

associated with an accumulator $H(\pi^*)$ with the highest value, compared with its neighbouring cells and with a fixed threshold H_{th}:

$$\pi^* = \{\pi : H(\pi^*) > H(\pi) \text{ AND } H(\pi^*) > H_{th}, \\ \pi \in N_{\pi^*}\} \qquad (4)$$

where N_{π^*} is the set of neighbours of π^* and consists of all points (ρ, θ) belonging to a mask of dimensions $A\mathbf{x}B$ centred in π^*. More details about maxima detection can be found in [17,19].

B. Geometric relationships among DPs

In this phase, collinear segments (Fig. 3a) are detected as subclasses related to the same π^* maximum. Parallel segments (Fig. 3b) are detected as points belonging to subclasses associated with a pair of the maxima characterized by the same θ value, but different ρ values. In this case, the relational parameters are the distance between two edges i and k, computed as $\rho_p = |\rho_i - \rho_k|$ and the angular orientation Γ_p. Convergent segments (Fig. 3c) are detected in two steps: (a) local maxima π^* are ranked on the basis of their accumulator values (from the lowest to the highest), and for each subclass $l_k(\pi^*)$ attached to a maximum, a general label λ is assigned to the points $\mathbf{x}$ belonging to that subclass by performing an antitransformation; (b) the points $\mathbf{x}$ for which a conflict among their labels occurs are inspected and a decision is made on the basis of the class dimension. Local maxima (x_η, y_η) are also extracted from the image coordinate space. Such maxima correspond to convergence points. If a local maximum is associated with more than one label (e.g., segments i and j), then the related labels represent the set of lines converging to the point (x_η, y_η). The following attributes are computed and associated with each label λ:

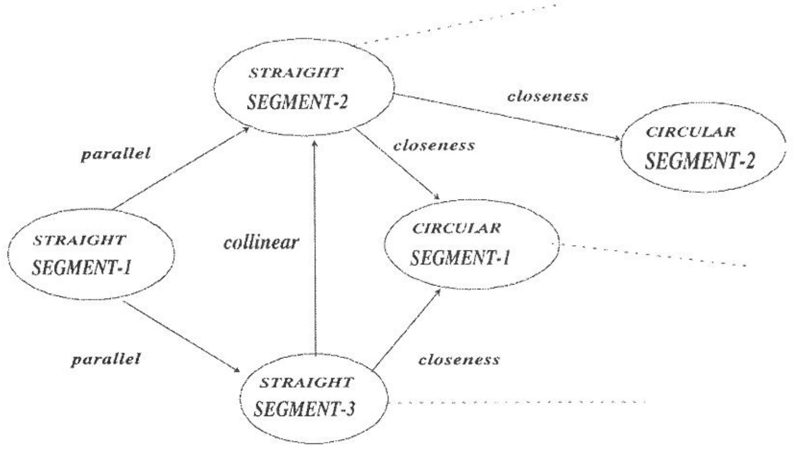


Fig. 4. Grouped-segment organized into the Descriptive Primitive Graph (DPG).

- the convergence point $(x_c, y_c) = (x_\eta, y_\eta)$;
- the convergence angle $\theta_c = |\theta_i - \theta_j|$;
- the weakness factor W_c;
- the maximum weakness factor $W_c^{\max}$.

Parameter W_c indicates the minimum distance of the convergent segment from the convergence point. If $W_c = 0$, this means that the segment intersects the convergence point. An analogous procedure is used for detecting circular segments (Fig. 3d) [17].

C. DP graph building up

Finally, straight and circular segments are organized into a graph (Fig. 4), defined as $DPG = \{s_m : m = 1, .., M\}$, where M stands for the graph dimensions (i.e., M is the number of detected segments) and sm denotes the graph node associated with the m-th segment. Geometrical relations among DPG nodes are established. Relations between two nodes are indicated as [12]:

$$R(k, j)=\{\mathbf{h}(s_k, s_j)=\mathbf{h}_{kj} : k, j\in[1..M],\ j\neq k\} \tag{5}$$

where $\mathbf{h}_{kj}$ represents a multidimensional array whose components are relational features of a segment pair. $\mathbf{h}_{kj}$ can be computed starting from the related pairs of intrinsic features $\mathbf{h}_{kj} = [status, \Gamma_{k,j}, \rho_p, x_c, y_c, \theta_c, W_c, W_c^{\max}]$ where the variable status can assume the following values {1 parallelism, 2 convergence, 3 collinearity}.

4. Grouping Module

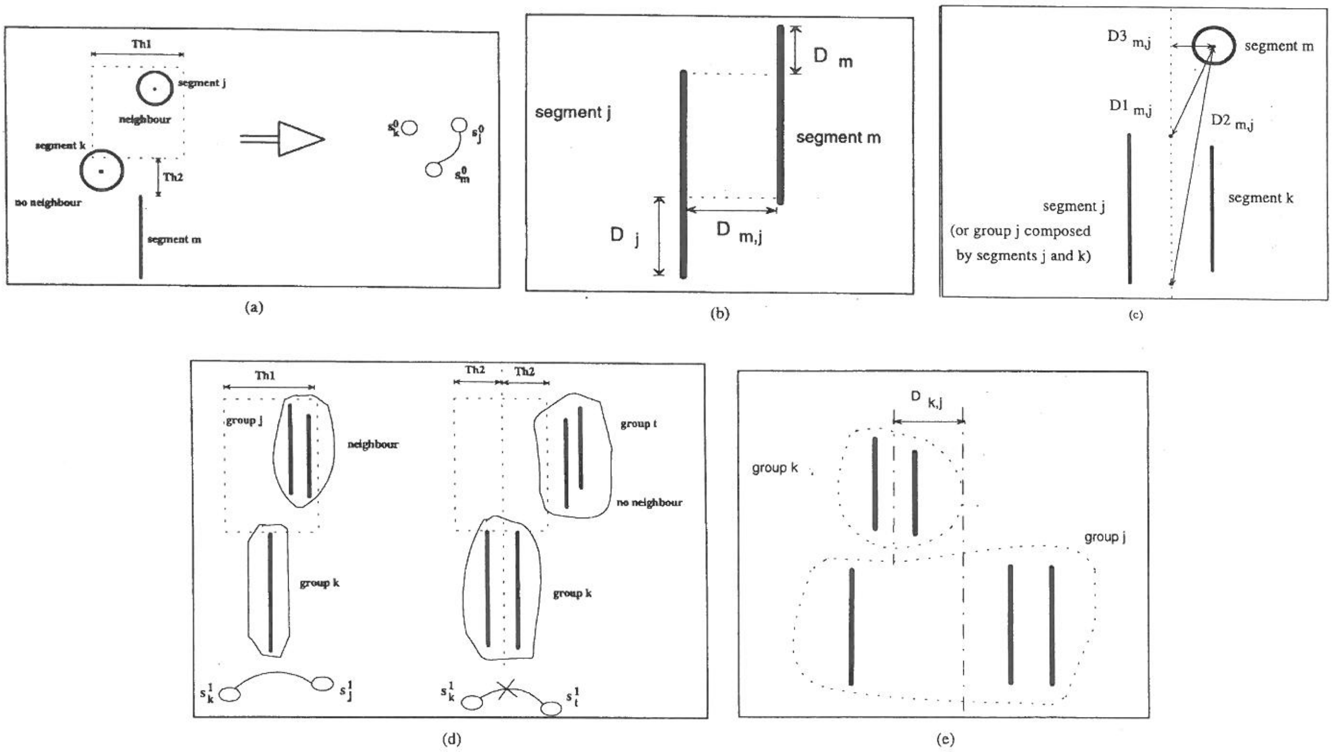
According to the kind of considered DPs, a non-homogeneous (i.e., varying from node to node) and multiple neighbourhood system has been defined. In particular, two lists of segments may be associated with each node: they specify the primitives that are related to a given node: parallelism, convergence, collinearity between straight segments and spatial closeness between straight and circular segments. Let $N_i^0 = \{N_{i,m}^0 : m = 1, .., M,\ i = 1, .., I\}$ be such a system, with $N_{i,m}^0 = \{s_k^0 : s_k^0 \neq s_m^0$ and $s_m^0 \in N_{i,k}^0\}$, (i.e., $I = 4$). Neighbourhood systems $N_{i,m}^0 (i = 1, .., 3)$ related to the parallelism, convergence and collinearity properties are provided directly by the segment extraction process as a list of segments parallel, convergent or collinear to the segment s_m considered. Neighbourhood system $N_{4,m}^0$ related to the vicinity property requires computation of the degree of closeness between rectilinear segments and circles. In particular, a circular segment is considered to be a neighbour of a rectilinear one if it is inside a window (of specified dimensions) close to the straight segment (Fig. 5a). Size and position of the window (with respect to the rectilinear segment) where the circle should be located are pre-defined by using a-priori knowledge about the object model considered.

The MRF process aims to associate features (i.e., rectilinear segments and circles) extracted by low level modules in order to detect different objects (e.g., road, cars, people) in the scene considered. It operates directly on the DPG, trying to group primitives, by associating a circle (the head) above one or more couples of close parallel vertical segments (legs, trunk) with a person, or two circles (wheels) under one pair of parallel horizontal segments (the lateral surface) with a car. To this end, the MRF process is split into two levels: the first is devoted to detecting various subparts of objects and the second level is devoted to merging groups of several parts detected by the first level to identify sets of DPs in a one-to-one relation with an object model.

A. First level

The first level of the MRF process, denoted as level 0, operates directly on the DPG nodes. The grouping process is performed by minimizing an adequate energy function composed of two terms related to the geometrical relations considered. In particular, the detection process of a person tries to assign the same label to close segments that are parallel and symmetric and below a circle. At the first level, the MRF model is specified by defining an irregular lattice $S^0 = \{s_m^0 : m = 1, .., M\}$, where each node s_m^0 is associated with a rectilinear or a circular segment, and a label field $R^0 = \{r_j^0 : i = 1, .., M\}$ defined on the same lattice S^0, where each r_i^0 is associated with a node s_m^0. Random values extracted from a set of M values are assigned to r_j^0, i.e., $r_j^0 \in [1, .., M]$. A vector g_m^0 of intrinsic properties is assigned to each graph node by the segment extraction algorithm, i.e., $G^0 = \{g_m^0 : m = 1, .., M\}$ is the observation field defined on S^0. Let Ω_0 denote the space of all possible configurations R_0. If R_0 is considered as an MRF with respect to the neighbourhood system $N_0 = \{N_m^0 : m = 1, .., M\}$, the best estimate R^{0^*}, according to the Maximum Probability (MP) criterion, is given by:

$$R^{0*} = \arg\min_{R^0 \in \Omega^0} U(R^0)$$
$$\text{with } U(R^0) = \sum_{c \in C^0} V_v(R^0) \tag{6}$$



(a) (b) (c)

(d) (e)

Fig. 5. Representation of (a) the neighbourhood system for straight and circular segments, (b) the clique for parallel segments and (c) the clique for the closeness property between straight and circular segments. (d) Definition of the window used to check if a generic node sj is a neighbourhood of a node sk and (e) representation of the parameter Dk,j.

where C^0 is the set of cliques associated with a neighbourhood system related to each node s^0_m, and V_c represents the potential function related to different cliques c. A clique is defined as a set of lattice sites (nodes), such that all the sites that belong to C^0 are neighbours of each other. In this case, the neighbourhood systems are provided directly as two lists of DPs that may be associated with each node, and that specify semantically different geometrical relations among nodes. Hence, it is necessary to define C^0_i as a set of cliques related to different neighbourhood systems N^0_i. Potential functions for each clique configuration are derived from a set of parameters P that allow one to discriminate between different situations in the same clique, i.e., $V_c = V_c(R^0, P^0)$. Therefore, the best configuration for the field R^0 can be determined by minimizing the following energy function:

$$U^0(R^0) = \sum_{n \in S^0} \sum_{i=1,2} \sum_{c \in C^0_i} V_c(R^0, P^0_c) \qquad (7)$$

where $P^0_c \subseteq P^0_i$, and P^0_i is the set of parameters that are dependent on the observation field G^0, and that represent different geometrical relations among the nodes considered in N^0_i. In particular, a parallel clique favours configurations of aligned parallel segments, i.e., the more symmetric a couple of segments, the more likely an edge configuration is to be accepted as a single group (see Fig. 5b).

A circle is grouped with a single rectilinear segment or with a pair of parallel segments, according to a proximity criterion, i.e., both the circle (located inside the window) that is closest to and most aligned with a rectilinear segment and the segment itself are very likely to belong to the same group. In general, energy functions for the relations considered are computed on the basis of a first-order neighbourhood system (i.e., cliques containing only two nodes). The energy function $U^0(R^0)$ for parallelism is expressed as:

$$\begin{aligned} U^0(R^0) &= \sum_{m \in S^0} \sum_{i=1,2} \sum_{c \in C^0_i} V_c(R^0, P^0_i) \\ &= \sum_{m \in S^0} \sum_{i=1,2} \sum_{j \in N^0_{i,m}} V^i_{m,j}(R^0, P^0_i) \qquad (8) \end{aligned}$$

where $V^i_{m,j}(.,.)$'s are potential costs related to the configuration composed of the current site s_m and its neighbour s_j. In particular, the clique potential function for the parallelism property is:

$$W^1_{m,j}(R^0, P^0_1) = \begin{cases} ps(P^0_1) & \text{if } r^0_m = r^0_j \\ K_1 & \text{otherwise} \end{cases} \qquad (9)$$

The function ps scores the degree of symmetry of a parallel straight line and is defined by the following expression (Fig. 5b):

$$ps(P^0_1) = D^2_m + D^2_j + D^2_{m,j} \qquad (10)$$

where D_m and D_j are relative displacements (with respect to the case of perfect symmetry) between the two edges, and $D_{m,j}$ is the distance between parallel segments. Therefore, function ps penalizes more strongly the grouping of segments that are not symmetric and that are far from each other. The clique potential function for the closeness property associated with one or a pair of (parallel) segments and a circular segment is defined as

$$V^2_{m,j}(R^0, P^0_2) = \begin{cases} vs(P^0_2) & \text{if } r^0_m = r^0_j \\ K_2 & \text{otherwise} \end{cases} \qquad (11)$$

Function vs scores the degree of closeness between a (couple of parallel) straight segment(s) and a circle, and is defined as follows (Fig. 5c):

$$vs(P^0_2) = \frac{D1_{m,j}}{D2_{m,j}} + D3_{m,j} \qquad (12)$$

where $D1$ and $D2$ are the distances between the centre of the circle and the extremes of the central axis related to the couple of segments (or the extremes of the single segment), and $D3$ is the distance between the centre of the circle and the axis itself. In this way, configurations composed of a circular segment close to the central axis of a pair of long edges are favoured. The resulting label image is made up of groups of primitives composed of a circle and a pair of segments, a circle and a single segment, and spurious groups of rectilinear segments.

The algorithm searching for the minimum-energy configuration of the field R^0 is a stochastic optimization method (i.e., Simulated Annealing [15] with the Metropolis sampler). According to this method, each graph node is iteratively considered, together with its neighbouring nodes related to different relational subsystems. The current field energy at iteration k, U_k, is evaluated. Then, the label of the current site is changed, choosing a new label among those of the neighbouring nodes, in a random way. The energy of the new configuration, say U_{k+1}, is computed and is either accepted or not, according to the Metropolis scheme [15], i.e.,

```
IF $U_{k+1} < U_k$ THEN accept
ELSE {generate a random number $A$ sampled from a
        uniform distribution in [0, 1]};
IF $A < \exp\{-(U_{k+1} - U_k)\}$ THEN accept ELSE reject
```

If the change is rejected, the previous label is reassigned to the node. Then, the next node, according to a predefined rank, is examined.

B. Second level

The second-level MRF process, denoted as level 1, operates on the feature sets formed by the first level, taking into account spatial relations among the detected groups. Hence, it assigns the same label to groups including only parallel straight segments, located between a group formed by parallel segment and a circle. A circle associated with more couples of parallel (vertical) segments identifying the legs and the trunk (or the arms) characterizes a person, and a circle associated with one or more couples of parallel (horizontal) segments identify the lateral body of a car. At this level, an irregular lattice is defined, $S^1 = \{s^1_k : k = 1, .., K\}$, where each node s^1_k is related to a group detected by the first level. A label field $R^1 = \{r^1_j : j = 1, .., K\}$ is also defined on the same lattice S^1, where each r^1_j is associated with a node s^1_k, and K is the number of groups detected at level 0. The final result is the best estimate of the label field R^1. Also in this case, a vector of observations g^1_k is assigned to each node s_k, i.e., an observation field is defined as $G^1 = \{g^1_k : k = 1, .., K\}$. G^1 is extracted from the final label configuration at level 0, i.e., $^G1 = F(R^0)$, where F is a function to be applied to R^0 to compute the parameters P^1.

At level 1, the neighbourhood system $N^1 = \{N^1_k,\ k = 1, .., K\}$ is unique and can be built by using, for each group s_k, a window of prefixed size, as at level 0. In other words, the node s_j is considered as a neighbour of s_k if s_j is contained in a space portion located over s_k (Fig. 5d). The size of the window is computed thanks to the a-priori knowledge about the application considered. At this point, following the considerations made for level 0, we can compute the energy function to be applied at level 1 as

$$U^1(R^1) = \sum_{k \in S^1} \sum_{c \in C^1} V_c(R^1, P^1_c) \qquad (13)$$

where V_c are the clique potentials related to possible configurations of the clique C^1, and P^1_c are the parameters involved in the cost computation. We can rewrite equation (13) by expressing the neighbouring system N^1_k as follows

$$\begin{aligned} U^1(R^1) &= \sum_{k \in S^1} \sum_{c \in C^1} V_c(R^1, P^1) \\ &= \sum_{k \in S^1} \sum_{j \in N^1_k} V_{k,j}(R^1, P^1) \end{aligned} \qquad (14)$$

where $V_{k,j} = \begin{cases} d_{k,j} & \text{if } r^1_k = r^1_j \\ K_3 & \text{othervise} \end{cases}$. In this case, parameters P^1 represent the axis coordinates of each group, and $d_{k,j}$ is the distance between the axes of the groups s_k and s_j (Fig. 5e). In this way, close groups are likely to be considered as a single group, until a standing person is detected. At this level, too, the minimum-energy configuration of the field R^1 is reached by using the Simulated Annealing [15] algorithm with the Metropolis sampler. There is no method used to determine optimal parameters K_i: they are fixed according to heuristic criteria, trying to find a good trade-off among the scale factors in different terms of the energy functions.

5. Geometric Reasoner Module

At the highest level, the Geometric Reasoner (GR) manages the 2D-into-3D transformation of DPs groups (called closures) into surfaces [5,13]. Only one of the closures provided by the lowest level is selected by means of an intrinsic matching phase in which the closure is assigned to one of the surfaces of the object to be searched for. The matching phase is mainly based on the similarity of an observed closure to the 2D shape of the model surface and it is performed by means of a fuzzy approach. The GR searches for one surface in each recognition cycle by considering model objects, which may consist of one or more surfaces. Once the GR has identified a surface, it does not perform additional grouping operations but a relational matching to verify whether the surface is geometrically compatible with the other surfaces previously found [13].

A. Recognition cycle

The GR considers an object model O_i composed of a set of surfaces:

$$O_i = \{SU_a : a = 1..A\} \tag{15}$$

The system has a geometrical representation of the object, given by the equations of the surfaces in the reference system of the model. In addition, the minimum and maximum dimensions of each surface are associated with each object model. For each object model, a viewpoint can be chosen from a subset of possible viewpoints. This selection (i.e., a viewpoint assumption) implies that the system must first select a certain transformation between the reference system of the object model O_i and that of the sensor on the basis of the a-priori knowledge on the most probable object pose. Selection of the viewpoint is very critical for functioning of the recognizer; it not only affects computation of the inverse transformation (once the viewpoint has been hypothesized), but also requires an explicit representation of different probable object poses.

B. Matching phase

When the viewpoints have been hypothesized, each 3D-object model surface belonging to a view can be transformed from the 3D-object model reference system into the 2D-image reference system by using the prospective projection equations [6,13]. To this end, a 2D-model closure is obtained for each model surface. As described in [13], each 2D-model closure is expressed by means of symbolic descriptions, associated with a set of 2D grouping properties (e.g., rectilinearity, convergence, etc.). These properties are defined by means of fuzzy membership functions [18] which are applied to different features in order to perform a numerical description of the 2D-model closure. The GR also contains information about the features to which fuzzy membership functions must be applied. Finally, if segment groups (e.g., closures) are available at the higher grouping level (i.e., the third level), appropriate fuzzy membership functions are applied to each group to perform numerical descriptions of the 2D real closures. Then, a matching function (MF) is applied to the numerical fuzzy values in order to detect the segment group closest to the characteristics of the 2D-model closure. For example, in the case of the object "road", described as a rectangular flat surface, one view must be selected among three possible views (i.e., one central view and two lateral views); the views are taken at a known height above the ground plane. If a central view is chosen, the 3D road model is mapped by a perspective camera model into a trapezoidal 2D patch, and

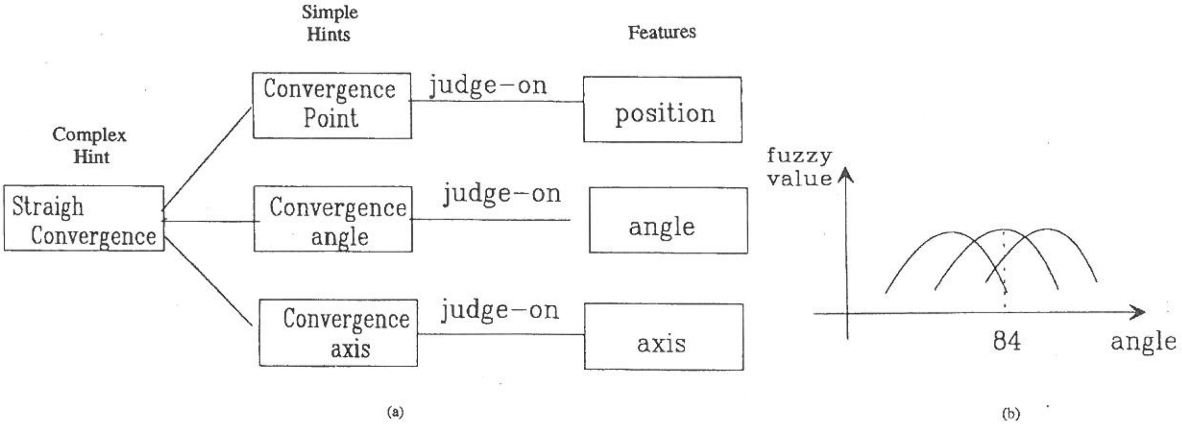


Fig. 6. (a) Example of how the "straight-convergence" fuzzy function is applied to the position (x,y) of the convergence point, to the convergence axis and to the convergence angle; (b) behaviours of the "straight-convergence" fuzzy function applied to the convergence angle from three different viewpoints.

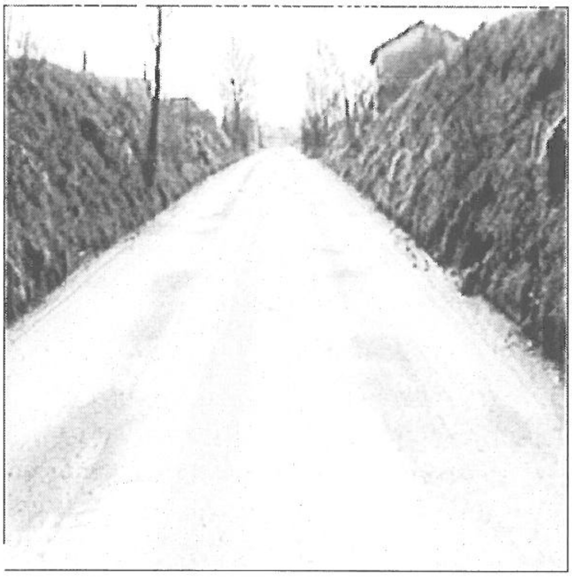


Fig. 7. Original road image without obstacles.

	CONVERGENCE POINT POSITION (X,Y)	*CONVERGENCE ANGLE*	*CONVERGENCE AXIS*
Right viewpoint	Right image position	small angle	right slanting axis
Central viewpoint	Central image position	medium angle	vertical axis
Left viewpoint	Left image position	small angle	left slanting axis

Table A The fuzzy description of the road model based on the position of the convergence point, on the amplitude of the convergence angle, and on the convergence axis equation.

a "straight-convergence" fuzzy function is applied to the different features used to describe the 2D closure [13]. Fig. 6a gives an example of how this fuzzy function is applied to the position (x,y) of the convergence point, to the convergence axis and to the convergence angle formed by the straight segments that make up the 2D closure. In particular, Fig. 6b shows the behaviours of the fuzzy function "straight-convergence" applied to the convergence angle formed by the straight edges, from three different viewpoints (i.e., one central view and two lateral views).

6. Results

The presented system was tested on about 100 images from a sequence acquired from a vehicle on a country road. Two examples of scenes were processed to assess the system's capability of detecting the road and the obstacles on it.

A. Road recognition

In the first image (see Fig. 7), an empty road is shown. A 3D model of a straight road is available at the Geometric Reasoner's level, and three possible viewpoints are considered: a central view and two lateral views. For each of these views, an expected rototranslation matrix is computed off-line. Figure 8 presents three possible viewpoints for the model of the road object. In Table A, the fuzzy descriptions of such hypothesized regions are given: such descriptions are based on the position of the convergence point, on the amplitude of the convergence angle and on the equation of the con-

	Group 1	*Group 2*
Right viewpoint	0.35	0.23
Central viewpoint	**0.95**	0.80
Left viewpoint	0.37	0.26

Table B Results of the matching between fuzzy models of the road object and the top-ranked group.

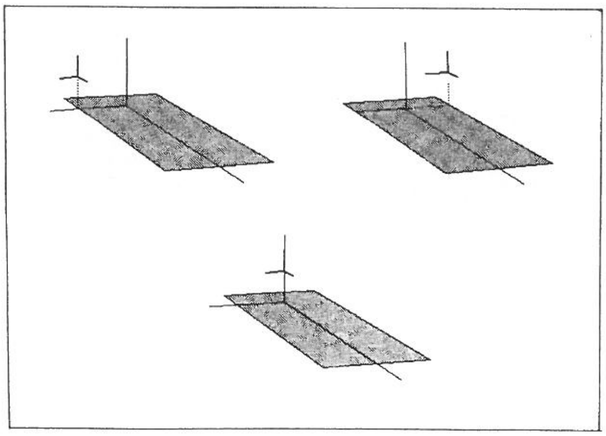
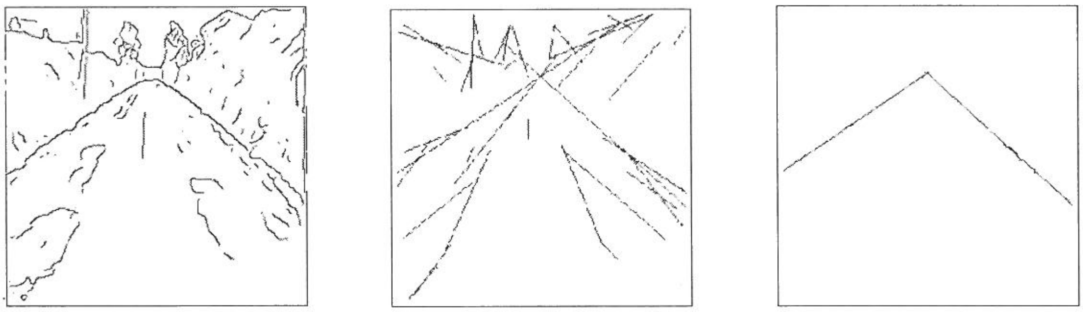


Fig. 8. Three different viewpoints for the object road.

	Parallel segment orientation	*Convergence angle*	*Parallel axis axis*	*Convergence axis*
Right view	(3 horizontal and 2 vertical segments) and (2 circles)	medium angle	(1 vertical axis) and (2 horizontal axis)	–
Frontal view	(4 horizontal and 4 vertical segments)	width angle	(2 horizontal axis) and (1 vertical axis)	1 vertical axis
Rear view	(3 horizontal and 4 vertical segments)	–	(1 vertical axis) and (2 horizontal axis)	–
Left view	(4 horizontal and 4 vertical segments) and (2 circles)	medium angle	(2 horizontal axis) and (1 vertical axis)	1 vertical axis

Table C Fuzzy description of the different car views



(a) (b) (c)

Fig. 9. (a) Edges extracted from the original image in Fig. 8, (b) straight segments extracted by the voting method and (c) results of the grouping phase.

vergence axis. The input image was processed according to the three-level process previously described. First, edges were extracted by means of the Canny algorithm, as shown in Fig. 9a. Then, straight segment were extracted by accepting the maxima above a threshold, $H_{th} = 20$ (Fig. 9b). Finally, the segments are grouped by using the MRF approach. In Fig. 9c, the results of the probabilistic grouping are presented. As one can see, the two straight segment of the road borders can be easily recognized. A ranking criterion based on the number and lengths of the segments inside a group was used to select the most interesting group to be compared with the models. Table B gives the results of matching the fuzzy models with the top-ranked group. The central view is selected as a verified hypothesis to be confirmed at the GR's level.

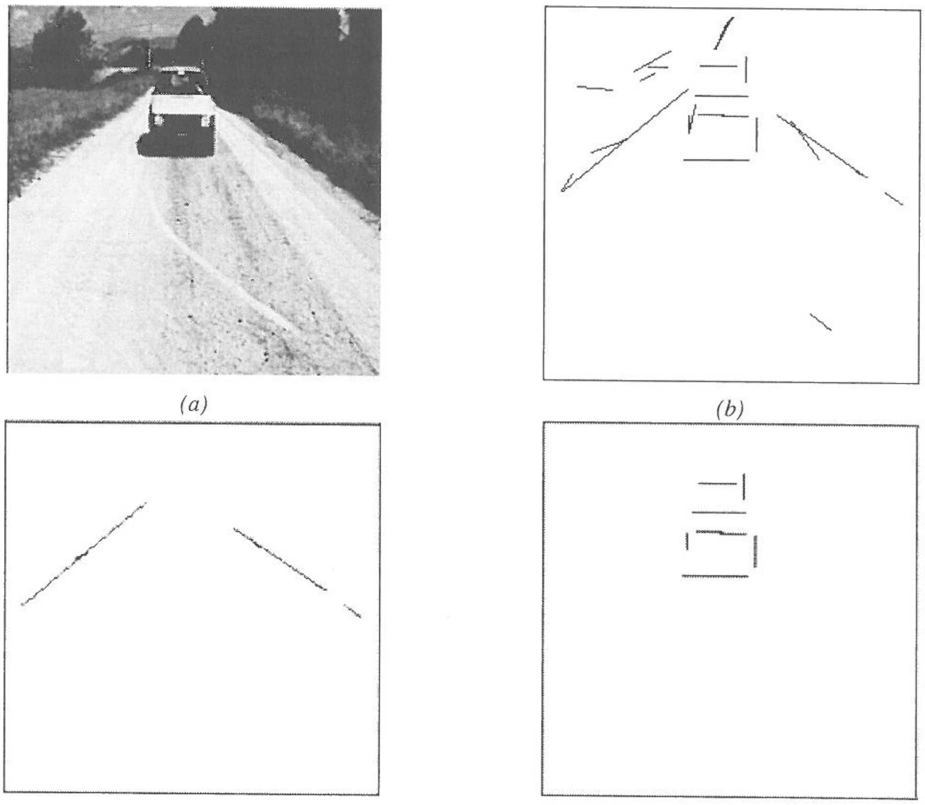
B. Obstacle detection and recognition

In Fig. 10a, a more complicated image containing a car (frontal view) placed in the middle of the road is shown. Detected straight segments are shown in Fig. 10b. No circular segments have been detected. The groups obtained by the MRF-based approach are shown in Fig. 10c and 10d, respectively. A 3D-car model and its significant 2D views are shown in Fig. 11. In Table C, the fuzzy description of the car views is given. The road model is the same as used for Fig. 8. As one can see, the groups receiving the highest number of votes can be associated with man-made or regular structures in the scene, i.e., the road which is characterized by long and convergent segments, and the car (frontal view) which is characterized by a pair of parallel segments contained inside the road region on the image plane. Results of the matching operation are presented in Table D, for the recognition of both the road and the car.

	Group 1 (road)	*Group 2 (car)*
Right (road)	0.20	0.11
Central (road)	**0.97**	0.20
Left (road)	0.31	0.13
Frontal (car)	0.07	0.35
Rear (car)	0.05	0.40
Lateral right (car)	0.12	**0.87**

Table D Results of the matching operation for the recognition of both the road and the car on the image in Fig. 10

Figure 12 shows another road scene containing a car (lateral view) placed in the middle of the road. Straight and circular segments are shown in Fig. 12b and 12c, respectively. Figures 12d and 12e show the detected DPs, which point out the best match with the road model (long and convergent segments) and the later view of the car model (two pairs of parallel segments). Finally, Fig. 12f shows the groups obtained by the MRF-based approach at the highest level: the complete shape of the car has been detected.


(a)

(b)

(c)

(d)

Fig. 10. (a) Real road image containing a stopped car (frontal view), (b) extracted straight segments and (c,d) obtained groups of convergent and parallel segments, respectively.

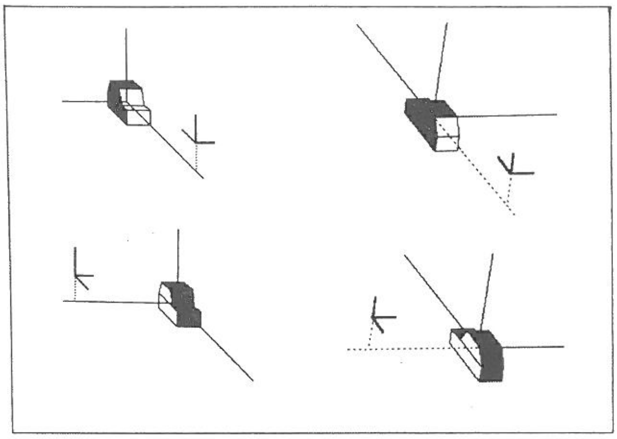


Fig. 11. 3D model of the car seen from four different viewpoints.

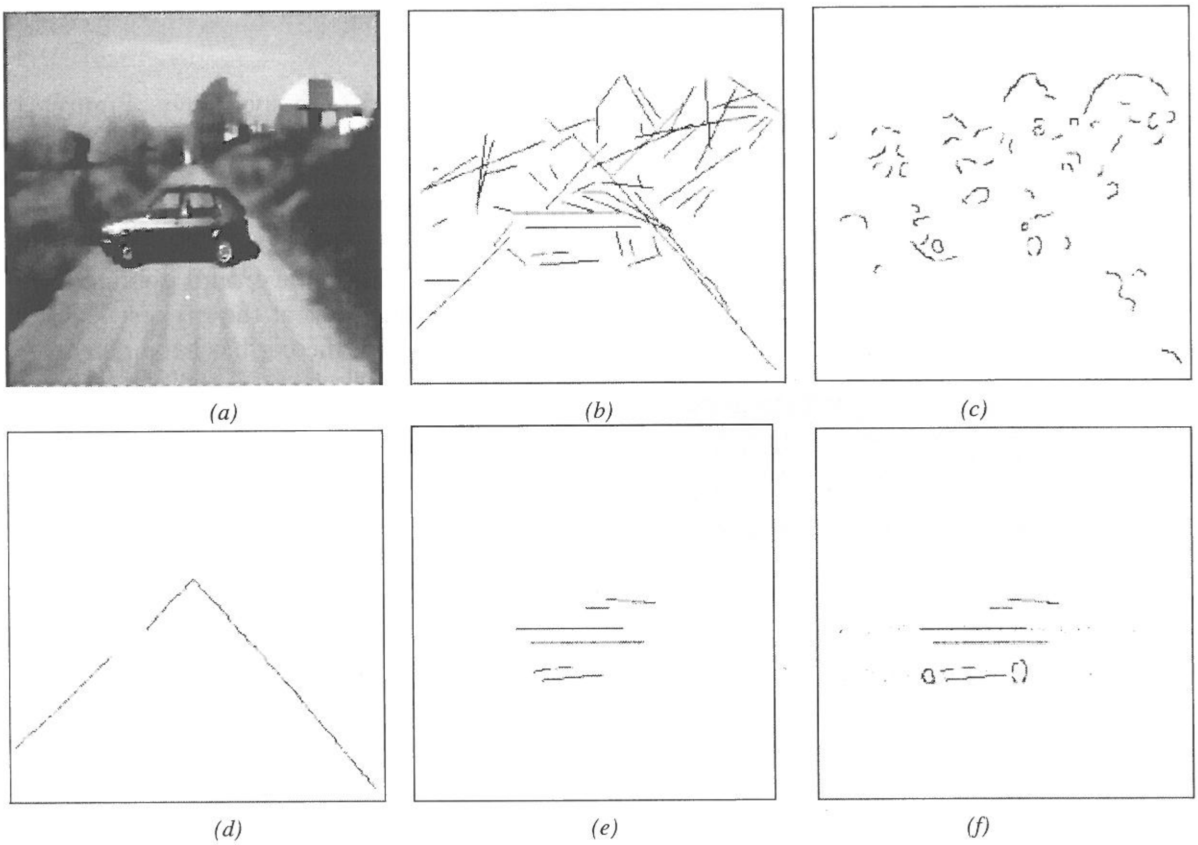


Fig. 12. (a) Real road image containing a stopped car (lateral view), (b,c) detected straight and circular segments, (d,e) obtained groups of convergent segments (matching with the model of the road) and of parallel segments (matching with the later view model of the car), respectively, (f) groups obtained at the second-level which match correctly with the model of the car.

A real scene containing more people (i.e., four, even the one in the bottom-left part of the image is almost completely out of the camera field of view) is used to give an idea of the capabilities of the system in classifying human obstacles (Fig. 13a). Figs. 13b, 13c and 13d show the edges extracted by the Canny algorithm, the rectilinear and the circular segments, respectively. Groups extracted by the first-level MRF process are presented in Fig. 14a. In particular, the heads, the legs and the bodies of three people are clearly detected. At the second level of the MRF process, such groups are merged according to the fusion rules of the object model to form the complete shape of a person (Fig. 14b). Vertical and parallel straight segments overhanged by a circle are searched for. As can be seen from these figures, despite the many primi-

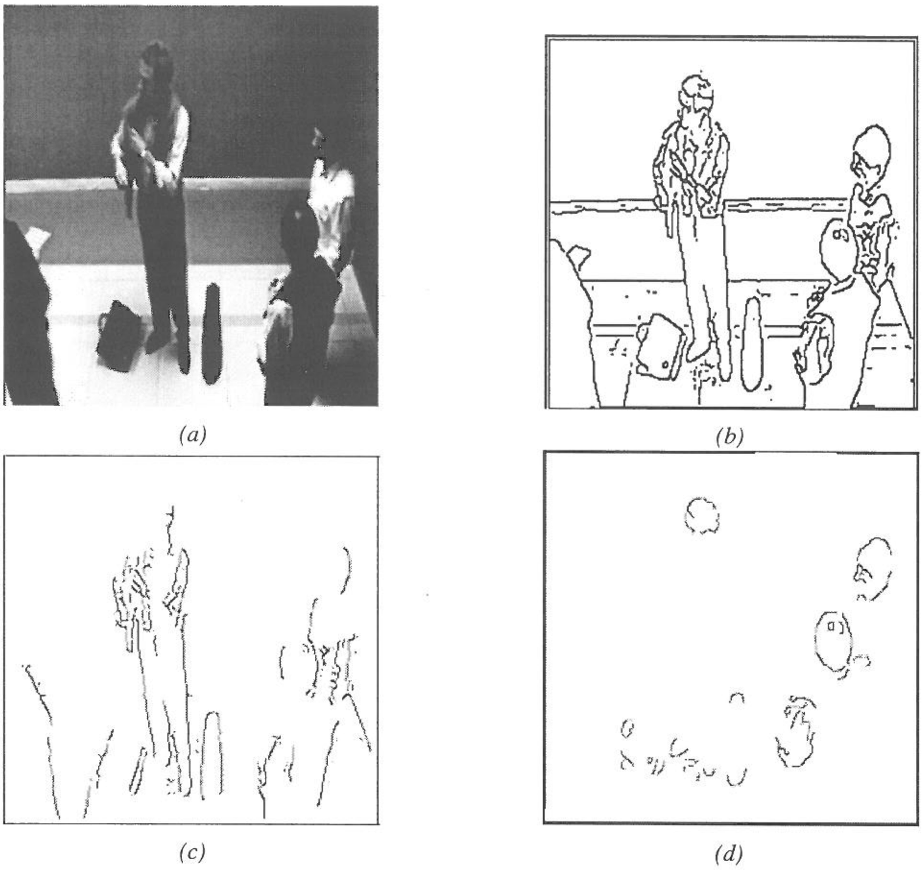


Fig. 13. (a) A real scene containing three people, (b) the edges obtained by the Canny algorithm, (c,d) the straight and circular segments, respectively.

tives detected at the first level, the MRF process is able to identify the groups associated with three persons. The fourth person has not been detected as the detected features (i.e., two vertical straight segments) are not sufficient to point out a correct match with the model of a person.

C. Time performances

The system was implemented on a SUN SPARC 20 Workstation. About 0.5 sec were required for the bottom-up processing, and the recognition of each object took about 2 seconds, including the formulation of multiple hypothesis and the matching operation. Implementation on multiprocessor hardware, to be used together with a pipelined image processor, is currently under development. Finally, Fig. 15 shows the CPU time required by the first level MRF process for different numbers of DPG nodes. As can be noticed, the time increases very rapidly for a number of nodes over 50, thus making the algorithm computationally too expensive. In general, scenes of moderate complexity involve a small number of nodes; therefore, computation time remains within acceptable limits.

7. Conclusions

In this paper, we have presented a visual recognition system for autonomous vehicle driving. The main novelty of the system, compared with the available ones [1-5], is an extensive use of grouping techniques to simplify the complexity of the data/model matching phase for object recognition. 3D models of the road and possible obstacles stopped on it have been considered. A new grouping algorithm has been proposed, which makes it possible to group together close groups of segments and circles. The algorithm is based on a Markov Random Field model of the descriptive-primitive graph, which works fast, mainly thanks to the reduced dimensions of the input space. This approach can be extended to other problems of similar type (e.g., surveillance systems for outdoor environments, etc.) by generalizing both the type of descriptive primitives represented in the graph and the neighbourhood relationships.

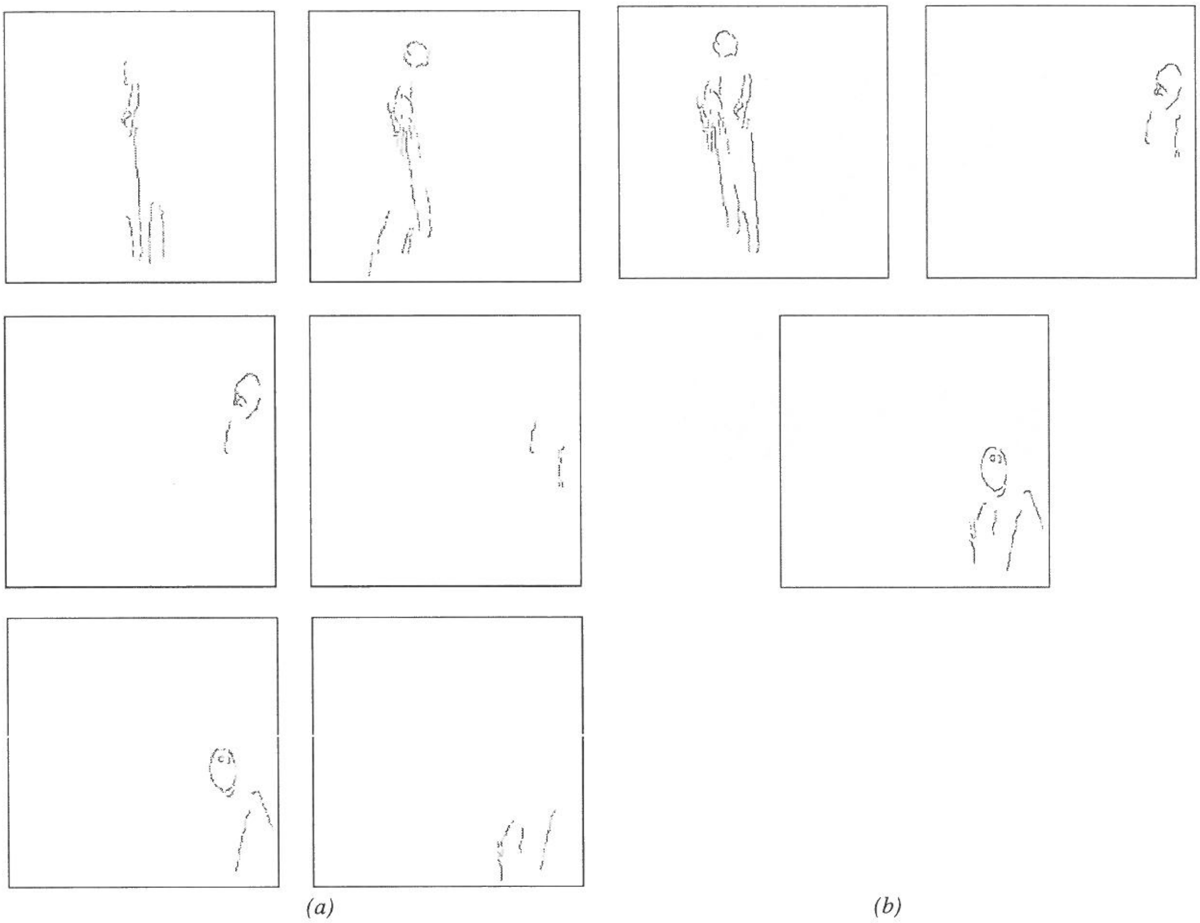


Fig. 14. (a) All groups extracted from the first-level of the MRF process and (b) groups obtained at the second-level which match correctly with the model of people.

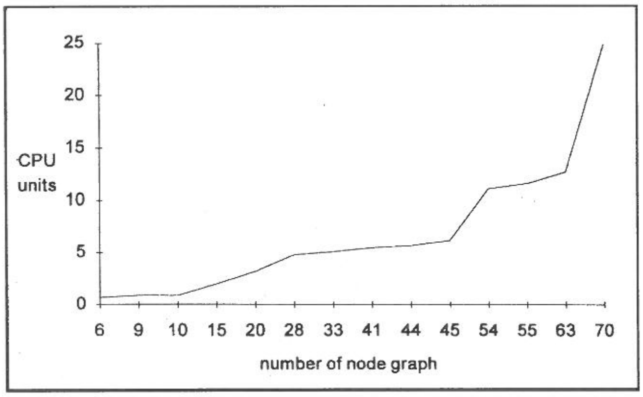


Fig. 15. Graph showing the CPU time required by the first-level MRF process versus different numbers of DPG nodes.

References

[1] E. D. Dickmanns, B. Mysliwetz, T. Christians, "An integrated spatial-temporal approach to automatic visual guidance of autonomous vehicles," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20, no. 6, pp. 1273- 1284, 1990.

[2] M. A. Thorpe, M. Herbert, T. Kanade, and S. Shafer, "Toward autonomous driving: The CMU Navlab," *IEEE Expert*, vol. 6, no. 4, pp. 31-52, 1991.

[3] M. A. Turk, D. G. Morgenthaler, K. D. Gremban, and M. Marra, "VITS - A vision system for autonomous land vehicle navigation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 6, pp. 342-360, 1988.

[4] N. KEHTARNAVAZ, N. C. GRISWOLD, AND J. S. LEE, "Visual control of an autonomous vehicle (BART) - The vehicle following problem," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 3, pp. 654-661, 1991.

[5] G. FORESTI, V. MURINO, C. S. REGAZZONI, AND G. VERNAZZA, "A distributed approach to 3D road scene recognition," *IEEE Transactions on Vehicular Technology*, vol. 43, pp. 389-406, 1994.

[6] P. V. C. HOUGH, "A method and means for recognizing complex patterns", *U.S. Patent* 3,069,654, 1962.

[7] R. O. DUDA AND P. E. HART, "Use of the HT to Detect Lines and Curves in Pictures", *Communication of the Association of Computing Machinery*, Vol. 15, No. 1, pp. 11-15, 1972.

[8] D. H. BALLARD, "Generalizing the Hough transform to detect arbitrary shapes", *Pattern Recognition*, Vol. 13, pp. 111- 122, 1981.

[9] D. G. LOWE, Perceptual Organization and Visual Recognition, Kluwer Academic Publ., 1986.

[10] D. P. HUTTENLOCHER AND P. C. WAYNER, "Finding Convex Edge Grouping in an Image", *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 406-412, June 1991.

[11] D. J. JACOBS, "A Grouping-Based Recognition System for Two Dimensional Objects", *Proc. IEEE Computer Society on Computer Vision*, 1987.

[12] G. L. FORESTI, V. MURINO, AND C. S. REGAZZONI, "Grouping as a Searching Process for Minimum-Energy Configurations of Labelled Random Fields", *Computer Vision, Graphics and Image Processing: Image Understanding*, Vol. 64, No. 1, July 1996, pp. 157-174.

[13] G. L. FORESTI, V. MURINO, C. S. REGAZZONI, AND G. VERNAZZA, "Distributed Spatial Reasoning for Multisensory Image Interpretation", *Signal Processing*, Vol. 32, No. 2, pp. 222-255, 1993.

[14] J. F. CANNY, "Finding edges and lines in images", *Artificial Intelligence Laboratory, Technical Report TR-720*, MIT, Cambridge, MA, 1983.

[15] S. GEMAN, D. GEMAN, "Stochastic Relaxation, Gibbs Distribuition, and the Bayesian Restoration of Images", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 6, pp. 721-741, 1984.

[16] G. L. FORESTI, V. MURINO, AND C. S. REGAZZONI, "A Markov Random Field Approach for Grouping of Intermediate Level Descriptive Primitives", *International Conference on Image Processing: Theory and Applications*, Sanremo, Italy, June 14-16, 1993, pp. 167-172.

[17] G. L. FORESTI, C. S. REGAZZONI, AND G. VERNAZZA, "Circular Arc Extraction by Direct Clustering in a 3D Hough Parameter Space", *Signal Processing*, Vol. 41, No. 2, 1995, pp. 203-224.

[18] L. A. ZADEH, Fuzzy Sets and Systems, North-Holland, Amsterdam 1983.

[19] G. L. FORESTI, "A Real-Time Hough-Based Method for Segment Detection in Complex Multisensor Images, *Journal of Real Time Imaging*, 1999, (in press).

Received: November, 1997
Revised: July, 1999
Accepted: September, 1999

Contact address:
Dr. Foresti Gian Luca
Department of Mathematics and Computer Science (DIMI)
University of Udine
Via delle Scienze 206
I-33100 Udine
Italy
Fax: +39 432 558406 phone: +39 432 558499
e-mail: `foresti@dimi.uniud.it`

GIAN LUCA FORESTI was born in Savona (Italy) in 1965. He received the laurea degree in Electronic Engineering and the Ph.D. degree in Computer Science from University of Genoa, Italy, in 1990 and in 1994, respectively. In 1994, he was Visiting Professor at University of Trento, Italy. Dr. Foresti is an Assistant Professor at the Department of Mathematics and Computer Science (DIMI), University of Udine, since 1995. Since 1998 he is Professor of the Artificial Intelligence course in the Science Faculty of the University of Udine. His main interests involve (a) artificial neural networks, (b) data fusion techniques for multisensor systems, (b) computer vision and image processing. The techniques proposed found applications in the following fields: automatic systems for surveillance and monitoring of outdoor environments (e.g., underground stations, railway lines, etc.), vision systems for autonomous vehicle driving and/or road traffic control, 3D scene interpretation and reconstruction. Dr. Foresti is author or co-author of more than 100 papers published in International Journals and Refereed International Conferences. Dr. Foresti has served as a reviewer for several international journals, and for the European Union in different research programs (MAST III, Long Term Research, Brite-CRAFT). He was Chairman and member of the Technical Committee at several conferences. He is member of IEEE, IAPR and GRIN.

CARLO REGAZZONI was born in Savona, Italy, in 1963. He received the Laurea degree in Electronic Engineering and the Ph.D. in Telecommunications and Signal Processing from the University of Genoa, in 1987 and 1992, respectively. He is the head of the Signal Processing and Telecommunications Research Group (SP&T) at the Department of Biophysical and Electronic Engineering (DIBE), University of Genova, that he joined in 1987. Dr. Regazzoni is Assistant Professor in Telecommunications at DIBE since 1995. Since 1998 he is Professor of Telecommunications Systems in the Engineering Faculty of University of Genova. Since 1997 he is Professor of Pattern Recognition at Master course in Information Technology at COGEFO, Milan, Italy (1997-2000) and since 1998 he is Professor in Telecommunications Systems at Diploma course in Computer Science at University of Trento, Italy.

His main current research interests are: Multimedia and Non-linear Signal Processing, Signal Processing for Telecommunications, Computer Vision for Video-Based Surveillance. He is a co-organizer of two Special Sessions on multimedia surveillance topics at International conferences (ICIAP99, Eusipco2000). Dr. Regazzoni has been project responsible in several EU research and development projects (ESPRIT P7809 DIMUS, P8433 PASSWORDS, P6068 ATHENA, AVS-PV, AVS-RIO)) and in several research contracts with Italian industries (Marconi Comms, Elsag S.p.A., Societa' AutostradeS.p.A.). He has submitted two patents concerned with Video Based Surveillance Systems for detection of abandoned objects.

Dr. Regazzoni is co-editor of the book "Advanced Video-based Surveillance Systems" (1999). He is author or co-author of 37 papers on International Scientific Journals and of more than 120 papers at Refereed International Conferences. Dr. Regazzoni has served as a reviewer for several international journals, and for EU in different research programs. He was Chairman and member of the Technical Committee at several conferences. Since 1997 he has been an expert in quality evaluation of High Level Education courses for CRUI under the CAMPUS project. He is a member of IEEE, and IAPR.