Feature selection for object tracking in traffic scenes

Sylvia Gil[1]
Ruggero Milanese
Thierry Pun

Computer Science Department
University of Geneva, Switzerland
E-mail: *gil@cui.unige.ch*

ABSTRACT

This paper describes a motion-analysis system, applied to the problem of vehicle tracking in real-world highway scenes. The system is structured in two stages. In the first one, a motion-detection algorithm performs a figure/ground segmentation, providing binary masks of the moving objects. In the second stage, vehicles are tracked for the rest of the sequence, by using Kalman filters on two state vectors, which represent each target's position and velocity. A vehicle's motion is represented by an affine model, taking into account translations and scale changes. Three types of features have been used for the vehicle's description state vectors. Two of them are contour-based: the bounding box and the centroid of the convex polygon approximating the vehicles contour. The third one is region-based and consists of the 2-D pattern of the vehicle in the image. For each of these features, the performance of the tracking algorithm has been tested, in terms of the position error, stability of the estimated motion parameters, trace of the motion model's covariance matrix, as well as computing time. A comparison of these results appears in favor of the use of the bounding box features.

Keywords: traffic scenes, motion detection, Kalman filter, tracking, feature comparison.

1. INTRODUCTION

Computer vision techniques can be useful in traffic control in order to increase safety and obtain road state information of monitored areas. For instance, the possibility to extract complex, high-level road information such as congestion, accident or fluid traffic allows to efficiently plan a path through the road network, to quickly bring rescue where needed or to deviate the traffic. In order to extract this type of information it is first necessary to segment moving objects from the scene. In this way, vehicles can be counted, and their trajectory, as well as their velocity and acceleration can be determined. Moreover, statistics can be collected from kinematic parameters in order to make a classification between safe, fluid, congestioned or dangerous state of traffic.

One of the major difficulties of monitoring traffic scenes, along with the real-time requirement, is the variety of light conditions of outdoor scenes. Indeed, the system should be reliable day and night, even though at night only vehicle lights are visible. Weather conditions also bring additional difficulties, such as the presence of the vehicle shadow in sunny days (shadows can prevent from correctly segmenting nearby vehicles) or a change in the contrast between the road and the vehicles when raining (a wet road is darker and generally dries irregularly). Thus, it is necessary to have a system able to adapt to these different lighting conditions by exploiting different visual features according to their reliability under such conditions. This paper presents a comparison of the ability of different features to be recovered and tracked, in an image sequence.

[1] Part of this research was conducted while the first author was a visiting scholar at the International Computer Science Institute, Berkeley, California, thanks to a grant from the Swiss National Fund for Scientific Research (4023-027036).

Surveillance of urban and highway scenes has been widely studied in the past five years, thus providing a large amount of literature. One of the most popular methods, called model-base tracking, uses a 3-D model of a vehicle and is structured in two steps: (i) computation of scale, position and 3-D orientation of the modeled vehicle, also called pose recovery, and (ii) tracking of the vehicle by fitting the model in subsequent frames by means of maximum-a-posteriori (MAP) techniques[1] or Kalman filters[2, 3]. The vehicle model being quite detailed (3-D model including the shadow), model-based tracking provides an accurate estimate (or recovery) of the vehicles 3-D position which might not be needed for most applications. A simplified model of the vehicle is proposed in [4] where it is represented through a polygon, with fixed number of vertices, enclosing the convex hull of some vehicle features. This model dramatically reduces the vehicle model complexity. In [4] Kalman filters are used in order to track the vehicle's position as well as its motion using an affine model which allows for translation and rotation. The fixed number of polygon vertices, however, allows little variations on the objects shape. Some improvements on this point are proposed in [5] through the use of dynamic contours instead of polygons with a fixed number of vertices. Cubic B-splines are fitted on a set of control points (vertices) belonging to the target and so providing a smooth parametric curve approximating its contour. In this case, a Kalman filter is used in order to track the curve in subsequent frames with a search strategy guided by the local contrast of the target in the image, i.e. with no use of the motion information. In the context of traffic scenes, especially in the case of highways, vehicle's motion should be a powerful cue in order to direct the search for the target position in subsequent frames. Another system that combines active contours model with Kalman filtering has been presented in [6]. In this case, the use of separate filters for the vehicle position and other motion parameters (affine model: translation and scale), has been shown to provide better results.

In consideration of this previous work, the approach described in this paper is based in the following points. First, advantage is taken from the simplicity of the targets profile (man-made vehicles), which can be well approximated by simple geometric models such as convex polygons; no restriction on the vertices number should be needed. Motion information in terms of an affine model (translation and scale) is used, as well as local contrast, in order to locate the vehicle in subsequent frames, by means of two separate Kalman filters. Finally, multiple features are tracked in the same image sequence and their performances are compared in terms of robustness, CPU time, and error measures. The rest of this paper is organized in the following way: Section 2 presents a motion detection system which discriminates between static background and dynamic objects and provides a set of binary masks coarsely representing the moving objects. Once moving objects are isolated, their mask shape is refined until their boundary accurately matches their contour (Section 3). After the mask refinement is accomplished, a set of features, such as the mask contour, the pattern describing the target itself, and its center of gravity, are computed for each vehicle, in order to be tracked in subsequent frames (Section 4). In section 5, the tracking procedure is described. Results are presented in Section 6, followed by a discussion. Finally, conclusions are presented in Section 7.

2. THE MOTION DETECTION SYSTEM

The goal of the motion detection module is to perform a segmentation between static and dynamic regions in an image sequence by providing a set of binary masks which coarsely represent the shape and the position of the moving objects. The method is required to be fast since it represents a preprocessing step for motion computation and tracking. For this purpose, it operates on low-level data such as spatio-temporal derivatives or image differences rather than an optical flow information.

2.1 Related work

Motion detection has been studied in different contexts such as video coding, surveillance, or traffic control. Differential methods are based on the substraction of subsequent frames in order to get rid of the constant background and process only the moving regions of the image. An example of this method is described in[7]: after performing the difference between successive frames, a 2-D median filter is applied on the difference image in order to smooth the mask boundaries; finally small regions are eliminated. This strategy is strongly affected by the aperture problem, when moving objects contain large regions of uniform gray-level. In this case, part of these objects are considered static and the resulting masks, despite the median regularization, appear oversegmented. A related approach, called the background method, aims at reconstructing the background using the spatial and temporal derivatives. When an accurate approximation of the background is available, it is subtracted from each frame in order to enhance moving objects. The background image has to be updated to account for changing external conditions (e.g. clouds). An example of this method is given in [8] in which a Kalman filter is used to update the background image. This method requires a certain number of frames until a reliable background is available, but its adaptability is a very attractive feature.

Other methods such as [9] exploit motion coherence through MAP techniques in order to separate objects undergoing different motions. This method minimizes, through a deterministic relaxation procedure, an energy function which combines a regularization term and a measure of match between spatio-temporal derivatives and the motion assigned to each region. This method, however, requires the computation of motion parameters and therefore does not meet our requirements described above. MAP techniques have also been used in order to compute global thresholds that segment images into *static* and *dynamic* areas[10]. Global thresholds are first computed according to the noise probability density function (pdf) of the difference images. Since segmentation through global thresholding does not provide well-segmented masks, local refinements are then applied on this preliminary data, based on the MAP criterion. However, this method still leads to oversegmentation, i.e. to many isolated masks which are actually part of the same moving object. Also, a major drawback of MAP techniques is the large processing power they require.

2.2 Motion detection with multiscale relaxation

In contrast to the previous approaches, our method is based on the simple difference of subsequent frames and requires only two frames in order to provide satisfactory results (see[11] for more details). The aperture problem, at the basis of the oversegmentation artifacts, is solved by the use of a multiresolution pyramid $I^l_{x,y}(t)$, for each frame $I_{x,y}(t)$ of the input sequence. At each level l of the pyramid ($l = 0, \ldots, \log_2 \text{image\_size}$), first estimates of motions are obtained by computing temporal image differences:

$$D^l_{x,y}(t) = I^l_{x,y}(t) - I^l_{x,y}(t-1) \quad . \tag{1}$$

Local differences $D^l_{x,y}(t)$ provide two motion contributions, through their magnitude, and through the locations of sign changes. These two factors are locally combined together to form the first motion estimates $E^l_{x,y}(t)$ (see Figure 1.b). High-resolution levels of $E^l_{x,y}(t)$ have a better spatial localization, but may only yield information at the object boundaries. Lower-resolution levels help solve the aperture problem, by filling in the interior of moving objects having constant grey level.

Multiple-resolutions motion estimates $E^l_{x,y}(t)$ are combined through a coarse-to-fine pyramidal relaxation process. Its goal is to locally propagate the pixel values *horizontally* within each level, as well as *vertically*, across contiguous levels of the pyramid. The "horizontal" component consists of a diffusion process within each pyramid level, to fill in gaps and reduce noise. The "vertical" component of the relaxation process combines information at location (*x,y*) of level *l* with that at locations $(2x+i, 2y+j)$, $i, j \in \{0, 1\}$ at the higher resolution level *l*-1 of the pyramid. The updating rule of the vertical component is defined by a multiplicative factor $\gamma^l_{x,y} \cdot \Delta^l_{x,y}$, in which $\gamma^l_{x,y}$ is a scaling coefficient.

The increment $\Delta^l_{x,y}$ is defined as a function of the difference image $D^{l+1}_l(t)$. That is, if the value of $D^{l+1}_{x/2,y/2}(t)$ is smaller than a threshold ξ (proportional the estimated image noise), then $\Delta^l_{x,y}$ is the quadratic term f_1, and otherwise it is given by f_2 :

$$f_1 = -k_1 \cdot (D^{l+1}_{x,y} - \xi)^2 \ , \tag{2}$$

$$f_2 = g \cdot \left(D^{l+1}_{x,y} - k_2 \cdot \xi \right) , \tag{3}$$

where $g(\alpha)$ is a sigmoidal function of the type $1/\left(1+e^{-\alpha}\right)$, and k_1, k_2 are positive constants. This algorithm corresponds to pushing the values of the estimates $E^l_{x,y}$ further towards either 0 or 1.

After the application of this algorithm, the full-resolution image at the bottom of the pyramid contains a binary mask of the moving objects $M^0(x, y)$. Due to the diffusion component of the relaxation process, the shape of these regions tends to be "convex", and to adapt to the shape of the underlying objects. Figure 1 presents the results on the sequence "walking". Despite the shadows and other reflecting surfaces, the resulting masks correspond well to the shape of the moving object.

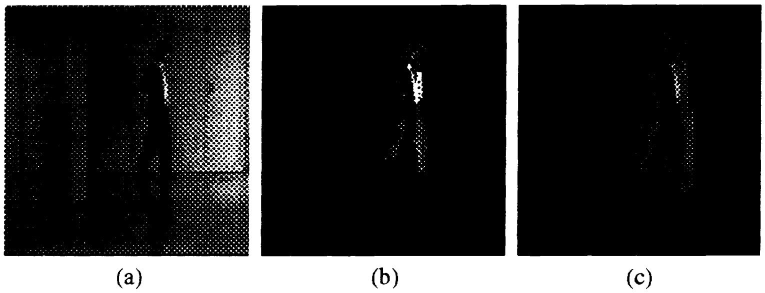

Figure 1: Results of the motion detection module on the "walking" sequence: (a) first frame of the sequence $I_{x,y}(t=0)$; (b) estimate $E^{l=0}_{x,\ y}$; (c) resulting binary mask superposed to the original image.

3. MASK REFINEMENT AND PROPAGATION

3.1 Mask refinement

The masks provided by the motion detection module represent a coarse description of the moving object, that must be refined. The strategy proposed in this section refines the initial mask boundaries according to the magnitude of spatio-temporal derivatives in the proximity of the initial mask. A similar approach has been adopted in the field of medical imaging for tracking contours of moving organs and cells. Several solutions of different complexity have been proposed. Geiger and Vlontzos [12] present a method to match the inner and outer boundaries of a moving heart wall, by defining a cost function to be minimized. In this context the two contours are known in advance. The cost function thus only takes into account a smoothness constraint on the motion field and a penalty factor for large unmatched arcs of boundaries, while ignoring the image intensity or gradient. Although this approach is not applicable in our context, (the final contour is not known in advance), the regularization terms apply in both situations. Leymarie and Levine [13] describe the tracking mechanism of moving cells by means of active contours (snakes). In their case, an initial snake is matched in the following image using a potential surface which takes into account the image intensity. Low-pass and band-pass pyramids of the potential surface are constructed in order to let the snake evolve while preventing local minima. Snakes are a suitable representation for objects that undergo non-rigid motion, and in the context of vehicles, some simplifications can easily be applied.

In the present work, advantage is taken of the geometric simplicity of vehicles by approximating their profile by a convex polygon in a similar way to [9] and [6]. The convexity assumption is not restrictive in the case of vehicles because in most projections their profiles are pretty compact and are thus well approximated by a convex polygon. This assumption considerably simplifies the matching step required by the tracking procedure, since it allows to by-pass problems such as contour regularization. Furthermore, an extensive literature exists on the topic of convex hull computation[14]. For each resolution level l, the binary mask $M_f^l(x, y)$ contains a number of regions $R_1,..., R_N$ representing moving targets. Due to the properties of the relaxation process, these regions tend to be slightly larger than the underlying objects. For this reason, the refining process uses each region R_i as a search window for the smallest convex polygon P_i containing the set of key points $\{\left(x_1^i, y_1^i\right), ..., \left(x_m^i, y_m^i\right)\}$. Each key point $\left(x_j^i, y_j^i\right)$ within a regions R_i is defined as a point where the spatio-temporal derivatives $D^l\left(x_j^i, y_j^i\right)$ exceed a certain threshold. These operations are actually limited to low resolution images in order to save computation time. Figure 2 shows the coarse initial masks issued from the motion detection algorithm (2.a) versus the contours of the final mask after the refinement process (2.b).

3.2 Mask propagation to higher resolutions

The propagation of the refined mask to higher resolutions is performed by iteratively repeating the projection procedure from the top of the pyramid down to its bottom until the full-resolution image is reached. A straightforward method for this projection procedure is the search of the maximum spatio-temporal gradient in a window defined by the 2x2 neighborhood: $(2x + i, 2y + j)$, $i, j \in \{0, 1\}$ where (x,y) is a polygon vertex at the coarse level. However, this method is not satisfactory, since the resulting contour at the lower level tends to slide to neighboring and even static contours of the image, thus deteriorating the quality of the mask shape. In order to avoid this problem, the search is limited to the window obtained by scaling the refined mask of level *l* to the higher resolution *l-1*. The search window thus will contain the spatio-temporal gradient limited to the regions of the refined mask of the higher level. Figure 2 (b) and (c) shows the results of the contour propagation through different resolution levels for different image sequences.

4. FEATURES TO TRACK

Once the target has been accurately isolated from the background, some features must be chosen, in order to be tracked over successive image frames. Several choices are possible for target features, such as the target's color, its contour or a pattern defining its spatial layout. Two tendencies appear to emerge in the existing literature, corresponding to a representation of the target's contour and on its description as a region. Both representations have advantages and drawbacks. Contour-based approaches[15] are fast, since they are based on the (efficient) detection of spatio-temporal gradients. Their major drawback is that the contours of an object in an image not always have a physical meaning. Indeed, contour extraction depends on the local intensity variation between an object and the background, so that changes in their relative intensity may cause a contour to disappear. This type of features is thus reliable only when the contrast between the target and the background is sufficiently constant.

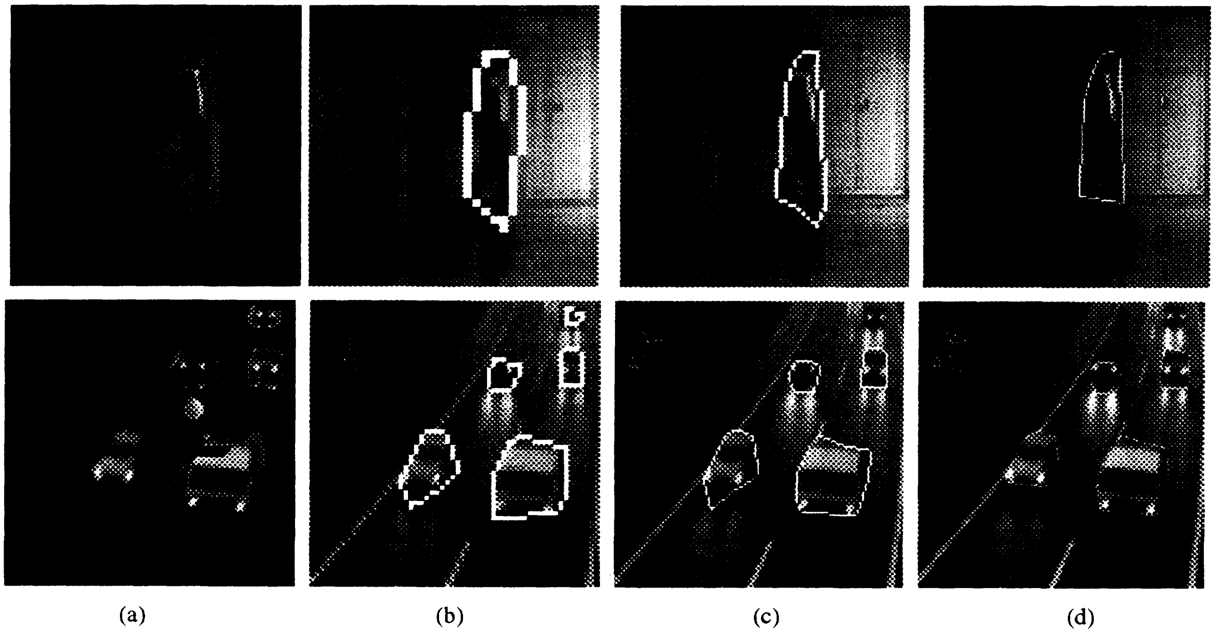

Figure 2: Mask refinement and propagation through the pyramid levels; (a) mask provided by the motion detection algorithm superposed to the one frame of the sequence; propagation of the refined mask (b) to middle (c) and high (d) resolution images.

On the other hand, region-based approaches[16] represent the target through a 2D-pattern; they are quite accurate and do not depend on the background. Their drawbacks are the computing time required for their manipulation (such as pattern matching) and the sensitivity of pattern matching techniques to changes in scale and rotation. Their use is most appropriate when the target size is small, when a low resolution approximation of the target is available or when other representations

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.