Online ISSN 1848-3380, Print ISSN 0005-1144
ATKAFF 57(1), 173–187(2016)

Amar El Maadi, Mohand S. Djouadi

Large-scale surveillance system: detection and tracking of suspicious motion patterns in crowded traffic scenes

DOI 10.7305/automatika.2016.07.881
UDK 004.932.72‘1:316.624; 621.397.48:656.021-052

Original scientific paper

The worldwide increasing sentiment of insecurity gave birth to a new era, shaking thereby the intelligent video-surveillance systems design and deployment. The large-scale use of these means has prompted the creation of new needs in terms of analysis and interpretation. For this purpose, behavior recognition and scene understanding related applications have become more captivating to a significant number of computer vision researchers, particularly when crowded scenes are concerned. So far, motion analysis and tracking remain challenging due to significant visual ambiguities, which encourage looking into further keys.

By this work, we present a new framework to recognize various motion patterns, extract abnormal behaviors and track them over a multi-camera traffic surveillance system. We apply a density-based technique to cluster motion vectors produced by optical flow, and compare them with motion pattern models defined earlier. Non-identified clusters are treated as suspicious and simultaneously tracked over an overlapping camera network for as long as possible. To aiming the network configuration, we designed an active camera scheduling strategy where camera assignment was realized via an improved Weighted Round-Robin algorithm.

To validate our approach, experiment results are presented and discussed.

Key words: Visual surveillance, Motion pattern, DBSCAN, Visual tracking, Crowded scene, Camera network, PTZ camera handoff, Planning

Veliki nadzorni sustav: detekcija i praćenje sumnjivih obrazaca pokreta u prometnim gužvama. Širom svijeta rasprostranjeni osjećaj nesigurnosti postavio je temelje za dizajniranje i implementaciju inteligentnih sustava nadzora. Velika upotreba ovih sredstava potaknula je stvaranje novih potreba analize i interpretacije. U ovu svrhu, prepoznavanje ponašanja i razumijevanje prizora postaju sve privlačnije povezane primjene značajnom broju istraživača računalne vizije, posebno kada se radi o vrlo prometnim prizorima. Analiza pokreta i slijeđenja ostalo je izazovno područje zbog značajnih vizualnih nejasnoća koje zahtijevaju daljnja istraživanja.

U radu je prikazan novi okvir za prepoznavanje različitih uzoraka pokreta, izoliranje neprirodnih ponašanja i njihovo praćenje pomoću nadzornog sustava prometa s više kamera. Primjenjuje se na gustoći zasnovana tehnika skupa vektora pokreta sastavljenih iz optičkog toka te uspoređenih s ranije definiranim modelima uzoraka. Neidentificirani skupovi tretiraju se kao sumnjivi i istovremeno su praćeni mrežom s više preklapajućih kamera što je duže moguće. S ciljem konfiguriranja mreže, dizajnirana je strategija raspoređivanja aktivnih kamera gdje je dodjela kamere ostvarena pomoću unaprijeđenog "Weighted Round-Robin" algoritma.

Ključne riječi: Vizualni nadzor, uzorak pokreta, DBSCAN, vizualno praćenje, prometni prizori, mreža kamera, PTZ prosljeđivanje kamere, planiranje

1 INTRODUCTION

There has been growing concern in designing and developing of smart computer vision systems. For security and safety applications, an ample evolution has been made in behaviors understanding and automatic anomalies identifying from complex and crowded video surveillance scenes. It requires to surmounting the computational cost and uncertainty challenging problems, with resolving computer vision difficulties including object detection, segmentation and tracking.

In crowded scenes, problems remain unsolved on how to get fine tracking capabilities against noise, occlusions and other visual ambiguities. For those reasons, we explore by this paper an innovative research axis toward reducing the dependence on the classical object segmentation and tracking. The preliminary version of the work appears in [1].

Furthermore, the challenges become increasingly harder with the increase in the scale and complexity of surveillance systems, mainly after camera networks advent. Structuring surveillance systems within network schemes may afford coverage improvement with more flexibility, but instigates problems such as camera handoff. It requires collaboration when performing different tasks. One of the most basic tasks is the objects tracking, which requests mechanisms to select a camera for a certain object and hand-off this object from one camera to another to seamless tracking success.

Selecting camera can be seen as an optimization [3] or as a scheduling problem [4], where the goal is to maximize the visibility of the features or events of interest while minimizing inter-camera switching. This has been an active area of research and many approaches have been proposed in both configurations: distributed and centralized. Some researchers bring hardware architecture design, some of which include embedded smart cameras, while others focus on the software design for camera assignment.

Rest of the paper is organized as follows. Section 2 discusses related works and Section 3 gives an overview of the proposed system. Section 4 illustrates the motion patterns modelling while Section 5 gives the recognition phase. In Section 6 we present the anomalies detection and tracking in an overlapping camera network. Results are discussed in Section 7 as Section 8 provides the conclusion and future work.

2 RELATED WORKS

Various research works have been accomplished to discover and understand abnormal behaviors of objects in video surveillance scenes. Conventionally, most of the proposed approaches turn on by subtracting moving objects from background images to deliver the motion information required for the tracking. In surveillance applications, this task is fundamental to improve lower level processing and higher level data extraction such as behavior analysis and recognition.

Tracking is difficult and challenging due to occlusions problems, environment conditions, noise and object complexity. However, the literature comprises a large variety of well-established algorithms. In [5, 6, 7], authors provided an interesting summary of literature surveys and developments related to visual surveillance systems, including motion detection, classification of moving objects, and tracking.

Alternatively, further solutions were proposed to improve patterns analysis and behavior understanding using features with no relies on tracking services. In [8], a framework is proposed to identify multiple crowd behaviors through stability analysis for dynamical systems, without the need for object tracking. Abnormal behavior detection is also studied in [9] using spatiotemporal cuboids, in [10] by modelling appearance and dynamics based on dynamic textures, and by learning normal behavior in [11].

The main review of [12] gave an update extending previous related surveys, and a focus on contextual abnormal behavior detection in video surveillance applications, which have involved the major interest in computer vision field.

Nowadays, by the growing need for video surveillance systems, the research spirit was focused on investigating large-scale solutions where camera networks fashion can play a key role. Multi-cameras systems offer various views of the same object at the same time, such that we can choose one or some of them to monitor an expected location. However, since multiple cameras may be involved over large-scale setting, we have to deal with the handoff problem.

Camera handoff is a decision process of switching from the current camera to another one to obtain a continuous tracking of the object of interest in multi-camera surveillance systems. This has been a hot topic of research and substantial number of approaches has been suggested in the literature.

Since camera handoff algorithms rely on tracking information, many works has concentrated on low and middle-level vision problems in the context of multi-camera surveillance systems. In [13, 14], color or other characteristic features of the interesting objects are matched, generating correspondence among cameras, while [15, 16] exploited homography to obtain position correspondences between overlapped views in the 2D image plane. Authors in [17] applied a modified mean-shift algorithm accompanied by a template matching method to real-time track multiple colored target objects using a PTZ active camera and a fixed passive camera. In [18], a scalable solution was recommended to the problem of tracking objects across spatially separated, uncalibrated cameras with non-overlapping fields of view.

Various other scheduling formulations and schemes were proposed, like in [19] where the scheduling problem was formulated as a graph-matching problem. Or else in [20], by creating a distributed lookup table to perform camera handoffs. The lookup table encodes the suitability of a camera to observe a specific location.

In [21] an approach to the problem of selection of optimal camera configurations in multi-camera systems has been newly presented. It includes a generalized statistical formulation of the problem, taking into account a set of user constraints, the number of cameras and the parameters of the cameras. A trans-dimensional simulated annealing algorithm has been designed to compute the optimal configuration.

Another system was designed to estimate the quality score for tracking PTZ active cameras [22], then with the scheduling algorithm to assign the cameras to track the best target object. Two trigger conditions are used to improve the planning flexibility. This work used three PTZ active cameras to monitor the experimental field, which different from the most of related work based on a set of fixed cameras and PTZ camera.

A valued work has been done by [4] as continuity of their previous papers [23, 24, 25]. They presented a proactive and deliberative planning strategy for intelligently managing a network of active PTZ cameras so as to satisfy the challenging task of capturing close-up biometric videos of selected pedestrians during their prolonged presence in a virtual train station environment under surveillance.

They formulated the multi-camera control strategy as an online scheduling problem and proposed a solution that combines the information gathered by the wide (FOV) cameras with weighted Round-Robin scheduling to guide the available PTZ active cameras, such that each pedestrian is observed by at least one PTZ camera while in the designated area.

The main objective of this paper is to investigate alternative representations and circumvent unreliable tracking via a camera network surveillance system. Our contribution can be situated in two levels. Firstly, through exploiting the motion analysis we propose a new architecture for an "*on-line*" use of density based clustering to suspicious motion patterns detection, recognition and tracking in crowded traffic scene. In the second part, we developed two different ways in which Round-Robin architecture was improved and appropriately made to be implemented for multiple suspicious vehicles tracking in an overlapping camera network. We accomplished an optimal planner that can performing the task of observing as many suspicious vehicles for as long as possible.

3 SYSTEM OVERVIEW

This section introduces the proposed framework for multiple suspicious motion patterns detecting and tracking in an overlapping camera network (Fig. 1). For crowded scenes application, low-level motion information is still required to afford high-level and useful semantic descriptions. This task involves modeling and appropriately classifying motion information with certain rules constrained by randomness and complex nature of crowded scenes.

In our application, we propose a new architecture for a two-phase system that can recognize different motion patterns and extract irregularities in a crowd traffic scene [1]. On the first phase (offline mode), dedicated for motion patterns learning and modeling, we use optical flow technique for computing motion vectors, then clustering them by DBSCAN (Density Based Spatial Clustering of Application

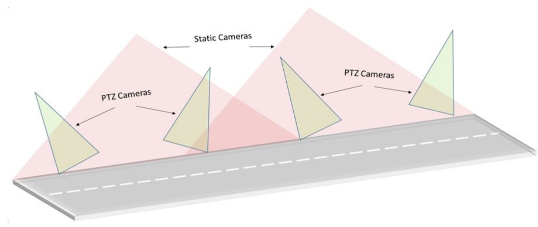


Fig. 1. View of the camera network scene

with Noise) algorithm [2], which allows to represent and model existing motion patterns.

Concerning the second phase (on-line mode) performed for identifying unusual motion patterns; we used an adapted DBSCAN algorithm that provides, at every frame, a number of noiseless clusters with arbitrary shapes according to objects density. By considering similarity and entropy criterions, resulting classes are matched to the motion pattern models. Unknown classes are carried out by multiple objects tracking module.

To expand the surveillance area and provide multiple view information, we were inspired from the works done by [4] to adapt and improve a weighted Round-Robin scheduling scheme to simulate a camera network model with overlapping fields of view to track suspicious vehicles during their spatio-temporal evolution in the scene. The camera network is populated with a set of wide field-of-view (FOV) passive cameras and active (PTZ) cameras. The wide FOV cameras are supposed calibrated with overlapping fields of view to insure the collect of useful data for detection and tracking in a large coverage area.

Besides, all suspicious vehicles should be obvious for optimum durations by the corresponding uncalibrated PTZ cameras before leaving the scene. Here, for each PTZ active camera we attribute a dynamic weight indicating its pertinence with respect to observing a vehicle. By this scheme, we expect an ample configuration for a realistic system that can automatically detect irregularities in a crowd traffic scene and keep tracking of multiple targets for a long time as possible.

4 MOTION PATTERNS MODELLING

Motion patterns modeling and validating are accomplished in "*off-line*" mode. As input, we exploit motion information from training videos frames containing significant patterns to be considered. The proposed architecture is shown in Fig. 2.

4.1 Motion vector extraction

This task is performed by computing dense optical flow using Gunnar Farneback's algorithm [26] from two subse-

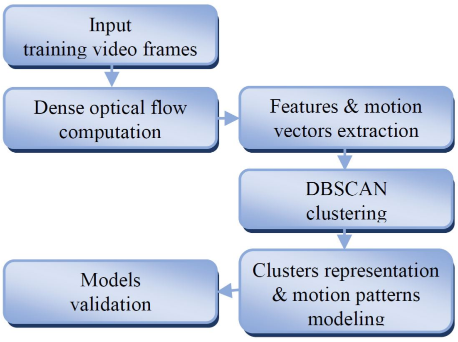


Fig. 2. Block diagram of the "off-line" mode

Fig. 3. Examples of motion vectors extraction using videos from four data-sets

quent frames. We implemented this algorithm via OpenCV (2.4.1) and we get fast and real-time results.

Optical flow applying returns a set of motion vectors (Fig. 3):

$$V_{i,t} = (x_{i,t}, y_{i,t}, m_{i,t}, \theta_{i,t}), \tag{1}$$

where $V_{i,t}$ is the motion vector i at frame t, represented by the feature point at the coordinate $(x_{i,t}, y_{i,t})$, the magnitude $m_{i,t}$ and the orientation angle $\theta_{i,t}$.

We get rid of noise by discarding vectors with small and large magnitudes. Significant motion vectors, computed over all training video frames, are used as an input to an afterward density-based clustering algorithm.

4.2 Motion vectors clustering

Among the many algorithms proposed in data mining field, DBSCAN is one of the most popular algorithms due to its high quality of noiseless output clusters. It discovers clusters with arbitrary shape with minimal number of input parameters.

The DBSCAN approach takes only two input parameters, epsilon (ε) and the number of minimum points (*MinPts*). *MinPts* is the minimum required points to form

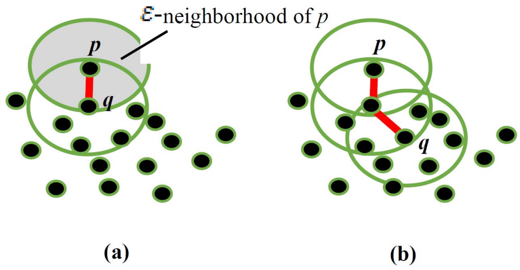


Fig. 4. (a) p is directly density-reachable from q (MinPts=6). (b) p is density-reachable from q (MinPts=6).

a core object, and ε is the distance between two objects to be considered ε-*neighbors* [1, 27].

The density associated with an object is obtained by counting the number of objects in a region of the specified radius ε around the object. An object with density greater than or equal to the specified threshold, *MinPts*, is considered as core (dense), otherwise non-core (sparse). Non-core objects that do not have a core object within the specified radius are discarded as noise. Clusters are formed around core objects by finding sets of *density-connected* objects that are maximal with respect to *density-reachability* (Fig. 4).

Thus, the algorithm does not take as an input the number of clusters to generate, it finds the optimum number of clusters based simply on the *MinPts* and the radius ε.

In order to determine the membership of each element to a cluster and make decisions we need a measure function. Minkowski distance is widely used:

$$d(i,j) = \sqrt[q]{|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + ... + |x_{in} - x_{jn}|^q}. \tag{2}$$

Many variants are used, mostly with $q = 1$ or $q = 2$. In our case, we have used $q = 2$, which represents the Euclidean distance.

Euclidean distance is still the most popular metric for similarity measure. It represents the shortest geographical distance between two points within a multidimensional space. This later space comprises as many dimensions as existing internal variables.

Hence, in case of three variables based classification, each participant will represent a point in a tridimensional space. Euclidean distance allows, so, the measure of the distance between those points to generate subsequent groups containing the neighboring points. Since it provides good performance, we have derived three variants:

- Variant a: using a simple difference between the position coordinates, the magnitudes and the orientations:

$$\sqrt[2]{|x_p - x_q|^2 + |y_p - y_q|^2 + |m_p - m_q|^2 + |\theta_p - \theta_q|^2} < \varepsilon, \quad (3)$$

where (x_p, y_p) and (x_q, y_q) are the coordinates of the points p and q; m_p and m_q are magnitudes of the motion vectors at the points p and q; θ_p and θ_q are the angles of the motion vectors at the points p and q, and ε is the threshold distance.

- Variant b: using the weighted Euclidean distance:

$$\sqrt[2]{\alpha_1 |x_p - x_q|^2 + \alpha_2 |y_p - y_q|^2 + \alpha_3 |m_p - m_q|^2} + \\ + \sqrt[2]{\alpha_4 |\theta_p - \theta_q|^2} < \varepsilon, \quad (4)$$

where α_i are the weights and

$$\sum_{i=1}^{4} \alpha_i = 1.$$

- Variant c: by dividing the distance into two parts [28], we added the magnitude information for better accuracy:

$$\sqrt[2]{|x_p - x_q|^2 + |y_p - y_q|^2 + |m_p - m_q|^2} < \varepsilon_m, \quad (5)$$

and

$$\min\left(|\theta_p - \theta_q| \leq \varepsilon_a, 2\pi - |\theta_p - \theta_q| \leq \varepsilon_a\right),$$

where ε_m represents a mixed threshold distance, and ε_a the angle threshold distance.

Using DBSCAN by one of these three distance measurements, we can cluster motion vectors into diverse groups of data points which have similar coordinates, similar magnitudes and similar orientation. Each group of motion vectors represents a motion pattern.

4.3 Motion patterns modeling

Based on the notion of density reachability, DBSCAN is designed to discover the spatial data clusters with noise. Here, clusters are populated by objects that can reach each other through densely populated regions.

To look for motion patterns, we cluster motion vectors resulting from optical flow process and we get groups of similar data points, as shown in Fig. 5.a.

To model a cluster, we proceed by fitting it's data using quadratic polynomial model (Fig. 5.b) and getting the three parameters (p_1, p_2, p_3) defined by:

$$y = p_1 x^2 + p_2 x + p_3. \quad (6)$$

Finally, we validate every model and represent every motion pattern with:

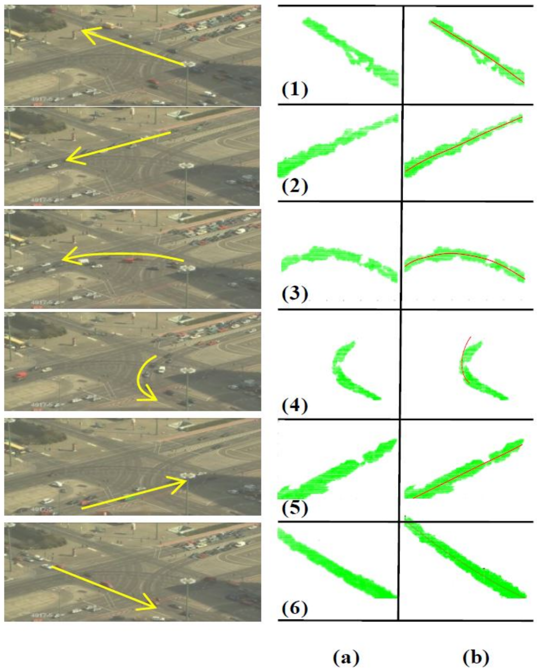


Fig. 5. Motion patterns modeling: (a) results of DBSCAN clustering, (b) motion patterns representation

Table 1. Motion Paterns Models

N°	P_1	P_2	P_3	θ_{mean}	θ_{std}	BW_{max}
Pattern 1	-0,00017	0,5488	90,83	24,170	7,5	80,62
Pattern 2	3,327	0,45055	-48,09	202,48	5,18	266,27
Pattern 3	0,00112	-0,7205	299,83	182,37	13,94	90,13
Pattern 4	0,00100	-1,586	693,80	141,73	18,29	347,63
Pattern 5	7,836e-05	-0,3913	256,38	162,99	7,89	139,01
Pattern 6	0,00019	-0,4655	423,40	-18,07	4,67	279,329

- the quadratic polynomial model presented by (p_1, p_2, p_3)
- the mean orientation angle (θ_{mean}) of the cluster and it's standard deviation (θ_{std})
- the maximum bandwidth containing the cluster ($BW_{\max}$)

Motion patterns models are illustrated in the Table 1.

5 MOTION PATTERNS RECOGNITION

In the previous phase, motion patterns were represented and modelled. In this step, the recognition process is performed at every frame (*online*) (Fig. 6).

At every frame, motion vectors are extracted after dense optical flow computation. Then we eliminate noisy vectors to refine results and reduce data to cluster.

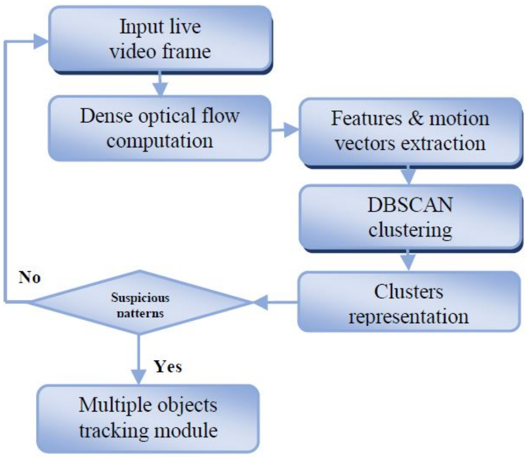


Fig. 6. Block diagram of the "off-line" mode

By DBSCAN algorithm, data points are populated into groups using the distance function previously defined. Once this algorithm is designed for mining large databases, it can be implemented to manage our data fairly fast for a "frame by frame" process.

5.1 Motion patterns representation and matching

DBSCAN provides several groups of similar data with arbitrary shapes and no prior knowledge about their number.

In order to compare newly furnished clusters, we have considered the combination of the two following aspects:

- maximizing the similarity inter-clusters; and
- minimizing the entropy inside a given cluster.

For best use of similarity measure between clusters, we have to opt for a good formalization [29]. To do so, every discovered cluster is compared with all predefined motion pattern models by the following formula:

$$Similarity = \alpha_1 B + \alpha_2 A + \alpha_3 M, \tag{7}$$

where:

$$\sum_{i=1}^{3} \alpha_i = 1,$$

B represents the minimal distance between the center of the mass of the new cluster and the representing curve of the motion pattern model. A is the distance between orientation angles, and M distance between magnitudes.

We have set empirically: $\alpha_1 = 0.4$, $\alpha_2 = 0.4$ and $\alpha_3 = 0.2$.

To throw away conflicts and be able to make better decision for the recognition, we have also introduced the entropy measure, which is a very important concept in information theory. Entropy can be defined as a measure of the expected information content or uncertainty of a probability distribution [30]. It can also be used to reflect the degree of disorder in a system or the uncertainty about a partition.

Entropy value ranges between 0 and 1. If this value is close to 0, it indicates the lower level of uncertainty, and the higher similarity in the sample. On the other hand, if entropy value increases adjoining 1, it indicates the higher level of uncertainty, the lower similarity in the sample. In our application, a cluster is assigned to a model if the entropy decreases after adding its data to the later.

5.2 Traffic density estimation

Describing and understanding the real traffic behavior require quantification of some of the basic traffic flow characteristics, such as speed, volume, or flow rate and density. In reality, density is difficult to measure from field as compared to volume and speed. However, it is considered the most important parameter of the three traffic-stream elements, because it is the measure most directly related to traffic demand.

Density is a typical variable from physics that was adopted by traffic science. It is defined as the number of vehicles occupying a given length of highway or lane and is generally expressed as vehicles per kilometer per lane. This is a critical parameter for uninterrupted flow facilities because it characterizes the quality of traffic operations. It describes the proximity of vehicles to one another and reflects the freedom to maneuver within the traffic stream [31].

In our application, traffic density was assimilated to the density of the cluster and defined as the number of data in an area unit. Therefore, we have defined the following three levels: high, medium and low traffic levels.

6 TRACKING IN AN OVERLAPPING CAMERA NETWORK

Performing an online motion patterns recognition scheme represents an added value to automated visual surveillance systems. On the other hand, we can prevail more if we take into account unusual apparitions and try to pursue their progression over time. Moreover, for safety and security applications, it is more interesting to detect and appreciate irregularities because fast decision-making is needed.

To this purpose, we present a simulation framework for a camera network populated with a set of wide field-of-view (FOV) passive cameras and active (PTZ) cameras

(Fig. 1). The wide FOV cameras are supposed calibrated with large but low-resolution coverage to insure the collect of useful data for detection and tracking.

Based on the output of the wide FOV cameras, the system generates spatio-temporal observation requests for all objects of interest, which are candidates for close-up views using PTZ uncalibrated cameras. However, such procedure can be more complex when the number of subjects and activities surpasses the number of PTZ cameras, especially for crowded scenes characterized by occlusions and irregular interactions. In this case, the scheduling and controlling of the PTZ cameras become problematic and more challenging.

To deliver close-up video of interesting objects during their presence in a scene, overall planning strategy for controlling active PTZ cameras should meet the following criteria:

- all objects of interest should be managed,
- no object should excessively favored over other objects,
- the capturing quality should be optimized, and
- assigning PTZ cameras to object should consider their capacities to meet the quality constraint.

To satisfy these requirements, we propose an adaptation and an improvement in the weighted Round-Robin scheduling scheme, performed by [4], with a particular interest in obtaining high-quality close-up videos of suspicious vehicles for performing number plate recognition or road-hogs identification tasks.

6.1 Suspicious vehicles tracking

Vehicle tracking is an estimation process, which comprises prediction and evaluation. Prediction includes a motion model that limits the search space for evaluation of vehicle presence in consecutive video frames. In our approach, the evaluation exploits color information and needs a comparison between the prediction of position and velocity, and the observed data.

Our tracking module was designed to deal with multiple simultaneous suspicious objects and execute tracking in an overlapping camera network scheme. This task can be released when an abnormal cluster is detected. To reject noise, only persevering cases are considered.

In case of wide field-of-view (FOV) passive cameras, we have implemented an autonomous module, which uses a background subtraction algorithm based on Gaussian mixture models. Morphological operations are applied to the resulting foreground mask to eliminate noise. Finally, Blob analysis detects groups of connected pixels, which are likely to correspond to moving objects.

To accurate tracking results, we have used multiple Kalman filters for every tracked object and incorporated color cue for associating detections over time. Every Kalman filter is also able to handle short-term occlusion by predicting its corresponding object motion.

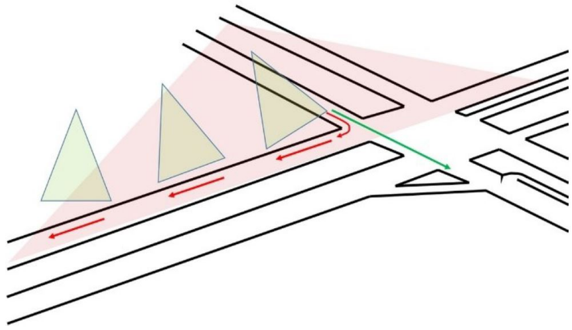


Fig. 7. Example of considered scenarios: the green arrow presents a "regular" motion direction whereas the red arrow presents a "non-regular" one. This last is considered as suspicious and tracked over the camera network.

Whereas, it is difficult to process tracking with active PTZ cameras due to the lack of a background model. Here, we opted for combining passive and active cameras in a master-slave setup [4], where detection and tracking information obtained from the fixed cameras is used to automatically control one or more nearby PTZ cameras.

6.2 PTZ Camera Scheduling

Camera scheduling is the core of our proposed multi-camera tracking system. The simulated environment consists of moving objects, static cameras, and active cameras. The static cameras are wide field-of-view (FOV) cameras that can only deliver low-quality information of the supervised scene. These cameras are supposed to be calibrated and can procure the 3D location, direction, and velocity information of the objects of interest.

The active cameras are PTZ cameras that can get high-resolution images of the objects of interest, representing the suspicious vehicles. We adapted a weighted Round-Robin scheduling scheme to create an optimal interaction between the active PTZ cameras and the scene, characterized by a crowded traffic, and provide a platform to choose optimal actions for these cameras in order to achieve high-quality close-up videos of suspicious vehicles.

Here, our implementation idea is that each object of interest should be tracked by a particular master camera. When the object nears the limit of this camera's field of view, neighboring PTZ cameras are put into slave mode, searching for the object until it appears. When a slave discovers the object in its field of view, it becomes the new master.

6.2.1 Weighted Round-Robin Algorithm

Weighted Round-Robin (WRR) is a preemptive algorithm that integrates priority scheduling. The structure for the WRR is similar to the Round-Robin (RR) in that it is simply a First Come First Served (FCFS) queue. Therefore, processes are dispatched in a first-in-first-out sequence but each process is allowed to run for only a limited amount of time. This time interval is known as a time-slice or quantum.

The WRR also uses the priority weight to assign the order of the tasks in the task queue. The task queue will be rearranged after a complete task queue cycle with the highest priority tasks requesting the processor time first (Fig. 8). This will enable high priority tasks to get the processor time first and always be at the beginning of the task queue. Once a task has completed, a new process is accepted in the queue. At this point, the tasks are reorganized in descending weight and the algorithm continues.

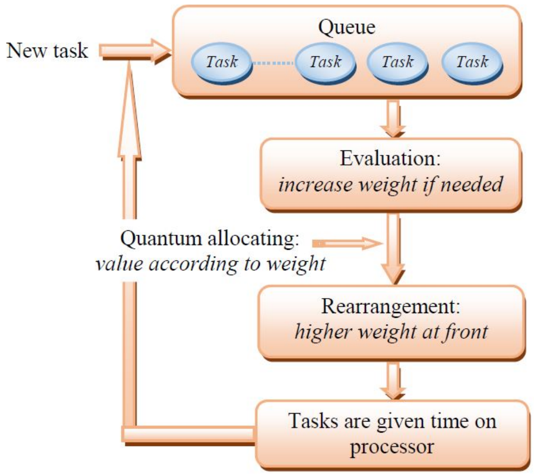


Fig. 8. Sequence of our WRR algorithm

As indicated previously, the scheduler needs a time management function to implement the Round-Robin architecture and requires a tick timer. The quantum is proportional to the period of clock ticks. The quantum length is critical issue in soft real time embedded application as missing of deadlines will have insignificant effects in the system performance. The quantum must not be too small which results in frequent context switches and should be slightly greater than average process calculation time.

6.2.2 PTZ Camera Relevance Computation

As improvement and adaptation of the camera relevance description given in [4, 32], we present the relevance of a camera to the task of observing a vehicle in terms of the following six factors:

- **Camera-vehicle distance** r_d: to favoring cameras that are closer to the vehicle.
- **Frontal viewing direction** r_γ: to favoring cameras having a frontal view of the vehicle.
- **In-scene sojourn** r_v: to favoring vehicles having fastest spatio-temporal progression.
- **PTZ limits** $r_{\alpha\beta\theta}$: Considering the limits of the PTZ camera parameters (the turn and zoom).
- **Observational range** r_o: reveals the observational constraints of a camera. It is set to 0 when the vehicle is outside the camera field-of-view; otherwise, it is set to 1.
- **Handoff success probability**r_h: to favoring handoff candidates neighboring the camera currently observing the vehicle. Factor r_o is considered only during camera handoffs; otherwise, it is set to 1.

Objects of interest in [4] were typically pedestrians without motion characterization. In our current application, we deal with vehicles abnormally behaving. We have introduced the "in-scene sojourn" factor (r_v) witch distinguishes between varied speediness in spatio-temporal progressions of those vehicles over the scene. When a vehicle arrives and moves with the highest speed, it will quit the scene after the minimum stay duration. Thus, the new factor (r_v) allows giving preference to the fastest vehicle because of its shortest sojourn in scene.

So, the camera weights with respect to a vehicle will be computed as:

$$r\left(c_i, h_j\right) = \begin{cases} 1 & \text{if } c_i \text{ is idle} \\ r_d r_\gamma r_{\alpha\beta\theta} r_o r_h r_v & \text{otherwise} \end{cases} \tag{8}$$

and

$$r_d = \exp\left(-\frac{\left(d-\hat{d}\right)^2}{2\sigma_d}\right), \tag{9}$$

$$r_\gamma = \exp\left(-\frac{(\gamma)^2}{2\sigma_\gamma}\right), \tag{10}$$

$$r_{\alpha\beta\theta} = \exp\left(-\frac{\left(\theta-\hat{\theta}\right)^2}{2\sigma_\theta} - \frac{(\alpha-\hat{\alpha})^2}{2\sigma_\alpha} - \frac{\left(\beta-\hat{\beta}\right)^2}{2\sigma_\beta}\right), \tag{11}$$

$$r_h = \exp\left(-\frac{(\varepsilon)^2}{2\sigma_\varepsilon}\right), \tag{12}$$

$$r_o = \begin{cases} 1 & \text{if } \alpha \in [\alpha_{\min}, \alpha_{\max}] \\ & \text{and } \beta \in [\beta_{\min}, \beta_{\max}] \\ & \text{and } d < d_{\max} \\ 0 & \text{otherwise} \end{cases}, \tag{13}$$

$$r_v = \exp\left(-\frac{(1/v)^2}{2\sigma_v}\right), \tag{14}$$

with: $\hat{\theta} = (\theta_{\min} + \theta_{\max})/2$, $\hat{\alpha} = (\alpha_{\min} + \alpha_{\max})/2$ and $\hat{\beta} = (\beta_{\min} + \beta_{\max})/2$. Where $\theta_{\min}$ and $\theta_{\max}$ are extremal field-of-view settings, $\alpha_{\min}$ and $\alpha_{\max}$ are extremal vertical rotation pan angles, and $\beta_{\min}$ and $\beta_{\max}$are extremal horizontal rotation tilt angles. α and β are, respectively, the pan and tilt gaze angles corresponding to the 3D position of the vehicle and θ corresponds to the field-of-view (zoom) setting needed to acquire close-up video of the vehicle.

The parameter d indicates the distance between the camera and the vehicle, $\hat{d}$ is the optimal distance between them while $d_{\max}$ represents the maximum distance for seamless tracking.

Parameter γ designates the angle between the fixation vector of the camera and the velocity vector of the vehicle, and ε is the angle between the fixation vector of camera c_i and the fixation vector of the camera currently observing the vehicle. The values of the variances σ_d, σ_γ, σ_θ, σ_α, σ_β, σ_ε and $\sigma_v$are empirically fixed as: $\sigma_d = \sigma_\gamma = \sigma_\theta = \sigma_\alpha = \sigma_\beta = 5$, $\sigma_\varepsilon = 45.0$ and $\sigma_\varepsilon = 0.5$.

Moreover, to avoid the situation where a PTZ camera can potentially track a vehicle for an uncertain duration, we have used the *preemption*; i.e., interrupting a running job to process another job, in the scheduling algorithm [25]. Choosing an appropriate value for the preemption cutoff time is essential issue when the presence of multiple objects of interest in the scene. A smaller value will increase cameras switching time between vehicles, whereas a too large value will implicate the reverse effect. It is recommended to choose value greater than the average camera assignment time.

To reach an optimum tracking duration of a maximal number of suspicious vehicles we built short-duration plans consisting of action/state sequences of lengths between 3 and 6.

7 EXPERIMENTS

In this section, we present an experimental study to evaluate and validate our approach using four data-sets containing videos of crowded scenes from real world.

Initial results are already shown in Fig. 3 where we have implemented a dense optical flow using Gunnar Farneback's algorithm. Time calculation were widely enough for a real-time request, and motion vectors were successfully extracted after noise removal action.

To evaluate DBSCAN clustering performance, we have used three variants of distance measurements defined by the equations (3), (4) and (5). We also have modeled and validated six motion patterns, as illustrated in Fig. 5.

Table 2. Results comparaison

Variant	Number of motion patterns	
	ground truth	*detected*
(a)	6	5
(b)		6
(c)		6

Results from the Table 2 compare the performance of the system using the three variants of distance measurements. We have registered a failure with the variant (a). The system made confusion between the motion patterns (1) and (3). As the variants (b) and (c) conferred more importance to the orientation parameter, the recognition tasks were successful.

Table 3. Results comparaison

Variant	Number of anomalies		
	ground truth	*good detection*	*false detection*
(a)	15	15	9
(b)		15	4
(c)		15	2

The Table 3 illustrates the ability of the system to detect anomalies in the supervised video scene. The ground truth contains fifteen anomalies counted manually (Fig. 9). In all variants the system could easily detect all the ground truth anomalies. However, some false detections were also considered as anomalies.

Errors were caused by noise resulting from the density based clustering. To overcome this weakness, tracking module was designed to track anomalies and strictly discard short time detections. The result of the multiple objects tracking is shown in Fig. 10. Here, we can notice that tracking task can be enabled only when an abnormal cluster is detected. To discard noise, only persevering cases are considered.

Examples in Fig. 11 demonstrate the three traffic density levels defined earlier: high, medium and low levels. Each cluster is presented by one color with three different concentrations, according to its density level.

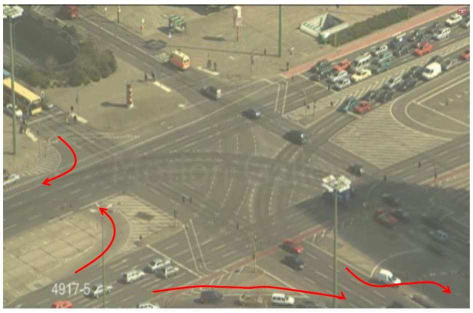


Fig. 9. Examples of motion patterns considered as not valid (red arrows)

For additional practical usage, the proposed approach can be used in traffic sign regularization systems deployed on road junctions. At a given interval of time, when the flow is only allowed on one or few lanes, we can robustly detect any anomaly or irregular motion on the other lanes (Fig. 12).

To expand our surveillance area and test our scheduling strategy in different situations, we have populated the simulated scene by 2 to 4 PTZ active cameras at each roadside around the road junction. We also have expected many scenarios implying one or more suspicious vehicles to be treated simultaneously.

In Fig. 14, we demonstrate the effect of the proposed WRR scheduling algorithm to enhance planning performance. Here, each colored segment represents a "*time unit*" for tracking duration of a given vehicle performed by one PTZ active camera. The colors are used to distinguish between diverse suspicious vehicles.

For this example, we attributed different velocity values to vehicle of interest designed by the black color, until it leaves the scene at *time unit* = 21. It speeds-up from Fig. 14.a to 14.e, and we can see changes in the planner conducts. The velocity value affects the relevance of the active camera to the task of observing our vehicle. While it can leave the scene faster than the others, the vehicle wins priority and will be tracked for more time duration.

Once we work with an overall planning strategy, we can also clearly see that the planner behaves differently with the other suspicious vehicles aiming an optimal control.

As we declared before, we are exclusively interested on tracking abnormally behaving vehicles over an overlapping network camera supervising a crowded traffic scene. To close realistic situations, we considered some scenarios comprising many simulated vehicles entering and leaving the scene at different times. In this part of our experimentations, the vehicles of interest move with identical velocity but in different directions, as presented in Fig. 18.

Figures 15, 16 and 17 illustrate the planning results via several configurations: using two, three and four PTZ active cameras. In each configuration, we have progressively augmented the number of interesting vehicles. After examining the whole results, we could distinguish between two cases: firstly, the number of interesting vehicles is equal or less than the PTZ cameras number. The second case when the number of those vehicles is superior to the number of the used cameras.

For the first case and in all our examples, the obtained results prove that the PTZ cameras have intelligently allocated their time among the different suspicious vehicles. Here, the planner worked perfectly and provided optimal results whatever the sense of the progression of each vehicles.

For the second case, which can rarely happens in real life; the number of suspicious vehicles exceeds the number of used PTZ camera. This situation becomes challenging since each PTZ camera can only track a single vehicle at every moment. As the number of the vehicles in the scene increases, the tracking module has increasing difficulty following the right vehicle. Hence, the planning process should accomplish a best time allocating to the task of observing the various vehicles in the scene.

Luckily, the shown results proved the significant improvement provided by our proposed solution to deal with the second case. The planner could dynamically use priority weight to assign the order of the tasks of observing multiples suspicious vehicles at the same time. Taking into account the velocity information has reduced the probability of losing interesting vehicles and considerably increased their observability time duration (Fig. 19).

8 CONCLUSION

In this paper, we proposed a new framework for an online use of density based clustering to motion patterns recognition and suspicious behaviors detection and tracking in crowded traffic scene. Beginning from a motion flow vectors generation, DBSCAN method was used to motion patterns modeling and recognizing.

We profited by the high quality of noiseless and arbitrary shape output clusters. The results show that our approach performances well on detecting motion patterns and anomalies in crowded scenes.

To close the real-world and give practical significance to the work, we tackled the problem of tracking in large-scale surveillance configurations. We proposed a simulation framework aiming the simultaneous tracking of a multiple suspicious vehicles over an overlapping camera network for as long as possible. The considered scene contained calibrated wide field-of-view passive cameras and

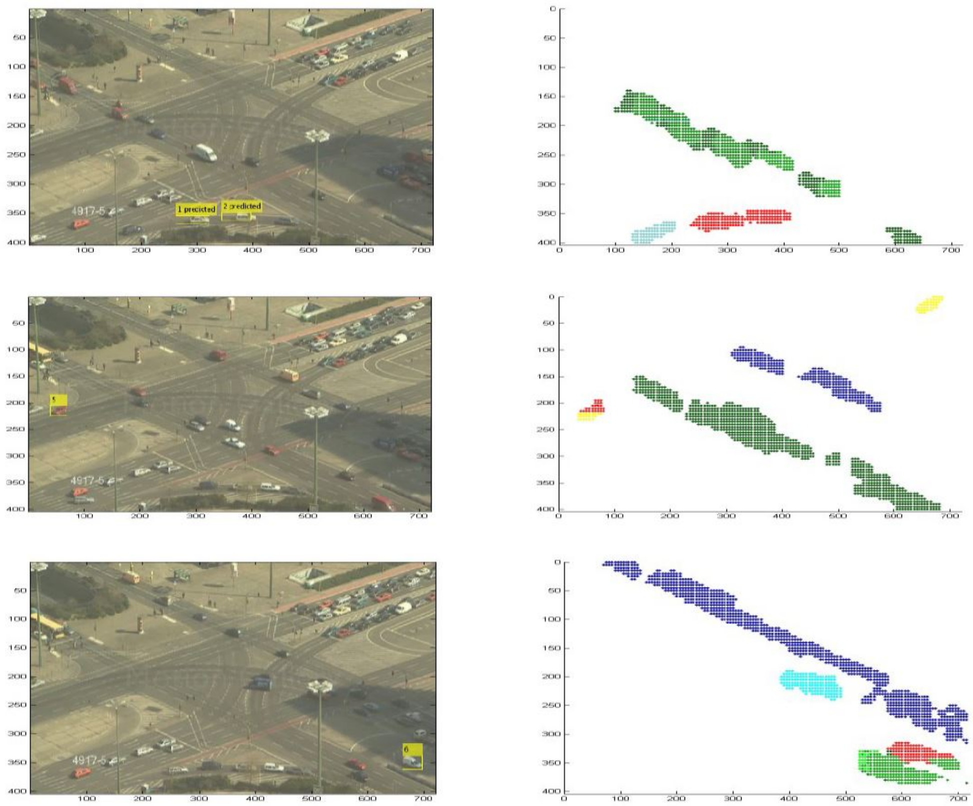


Fig. 10. Examples of anomalies detection and tracking

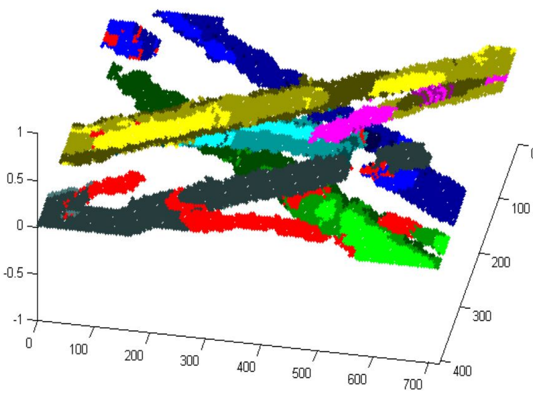


Fig. 11. Illustration of traffic density estimation

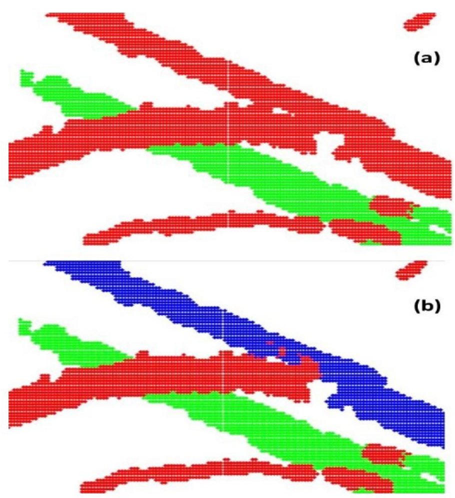


Fig. 12. Example of using the algorithm in traffic sign regularization: (a) only one motion pattern is allowed (green color); (b) Two motion patterns are allowed (green and blue colors)

uncalibrated active PTZ cameras. To manage, we opted for a scheduling strategy via an improved Weighted Round-Robin algorithm.

Round-Robin architecture was significantly improved and appropriately implemented to deliver solutions to the challenging problem of planning optimization when the number of tracked vehicles surpasses the number of used PTZ camera.

In the other hand, experimental results proved, as expected, an appreciable improvements in dynamic weight

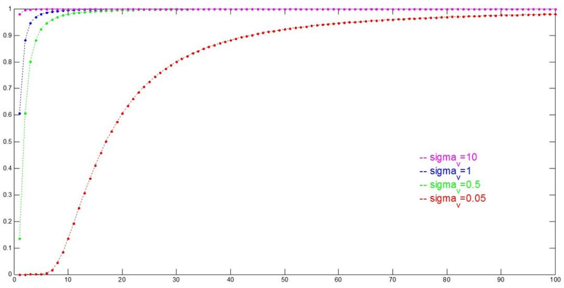


Fig. 13. Impact of the velocity on the relevance computation

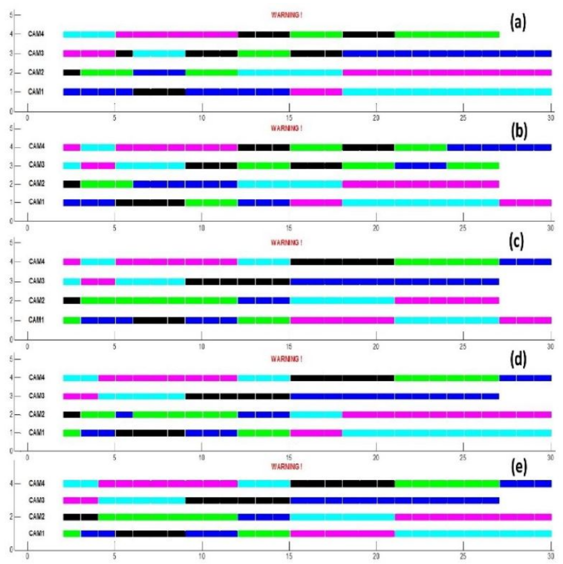


Fig. 14. Example of Performance enhancement via the improved WRR with four PTZ cameras and five suspicious vehicles moving in different directions: the vehicle designated by a black color segments speeds-up from (a) to (e) and gets more priority to be observed for more duration until living the scene at time unit = 21

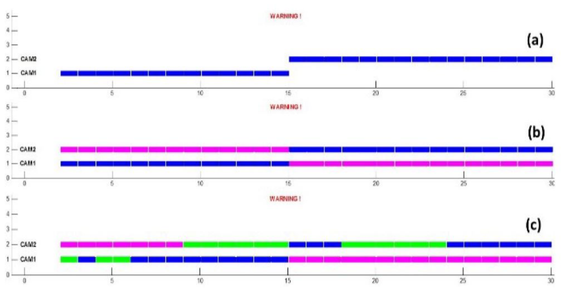


Fig. 15. Planning results using two cameras, the scene is occupied by: (a) one suspicious vehicle, (b) two suspicious vehicles, (c) three suspicious vehicles

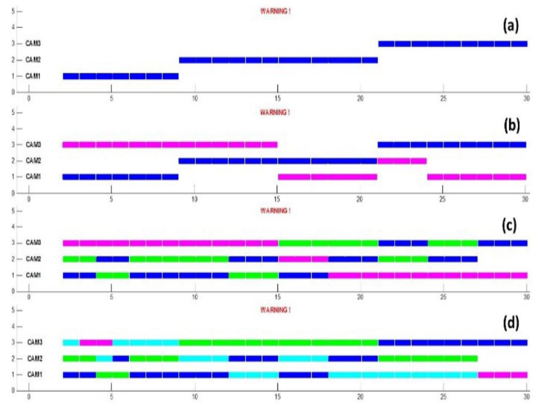


Fig. 16. Planning results using three cameras, the scene is populated by: (a) one suspicious vehicle, (b) two suspicious vehicles, (c) three suspicious vehicles, (d) four suspicious vehicles

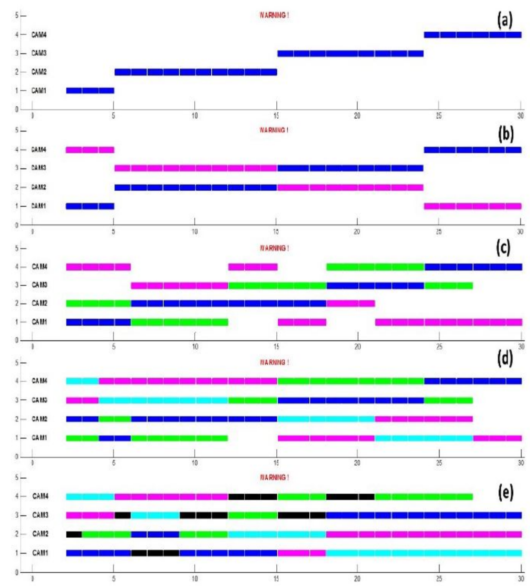


Fig. 17. Planning results using four cameras, the scene is populated by: (a) one suspicious vehicle, (b) two suspicious vehicles, (c) three suspicious vehicles, (d) four suspicious vehicles, (e) five suspicious vehicles

attribution, to each PTZ active camera, indicating its pertinence with respect to observing a vehicle of interest. The priorities were integrated according to progression speed of those vehicles in the scene.

REFERENCES

[1] A. El Maadi and M.S. Djouadi: "Suspicious motion patterns detection and tracking in crowded scenes", Proceedings. IEEE International Symposium on Safety, Security,

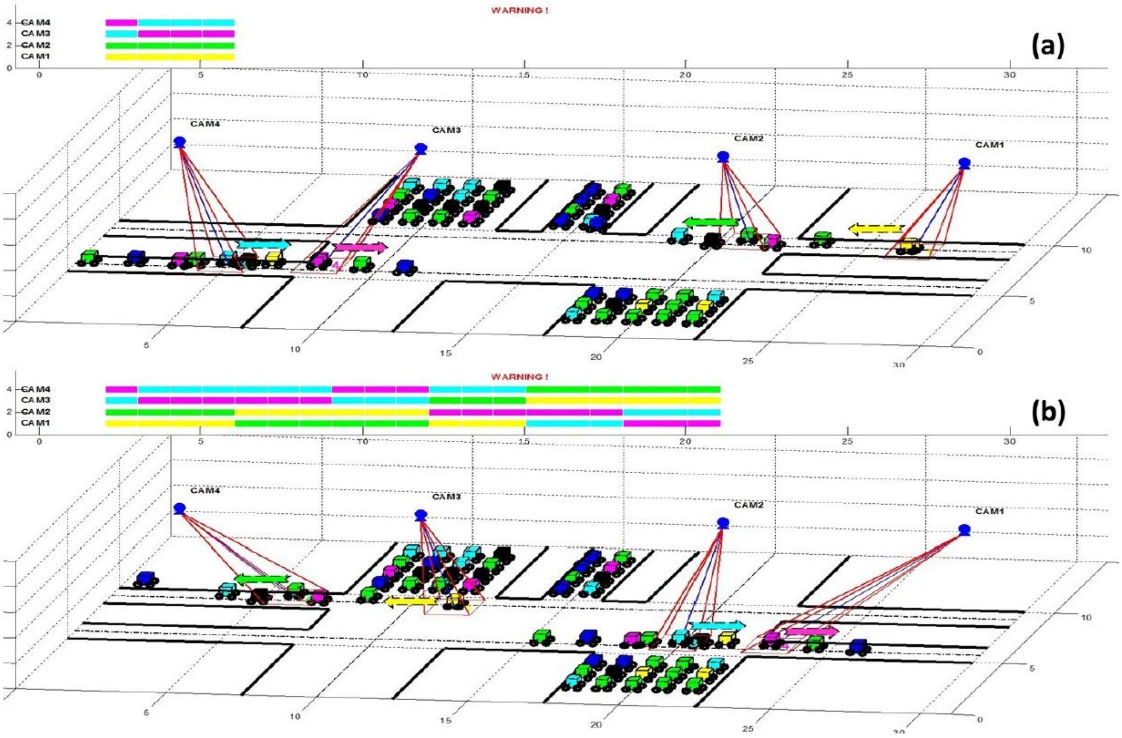


Fig. 18. A sample scenario containing four PTZ active cameras: from (a) to (b), the planner processes with four suspicious vehicles moving in different directions over the scene

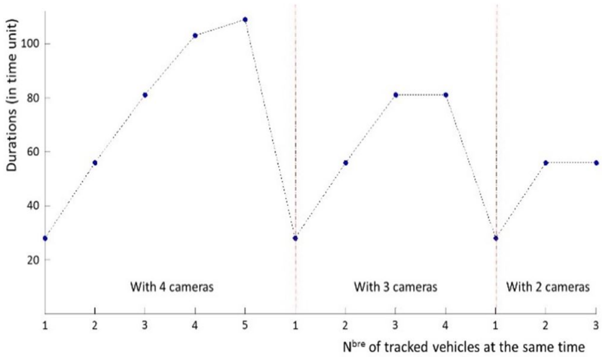


Fig. 19. Observability time duration (in time units), according to various planning configurations

and Rescue Robotic, Linköping (Sweden), October 21-26th, 2013.

[2] Ester M, Kriegel H.-P, Sander J, and Xu X: "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD'96), Portland: Oregon, pp. 226–231, 1996.

[3] H. Jiang, S. Fels, and J. J. Little: "Optimizing multiple object tracking and best view video synthesis", IEEE Trans. on Multimedia, vol. 10, no. 6, pp. 997–1012, Oct 2008.

[4] F. Z. Qureshi and D. Terzopoulos: "Planning ahead for PTZ camera assignment and handoff", in Proc. of ACM/IEEE Conference on Distributed Smart Cameras (ICDSC'09), Aug. 30-Sept.2, 2009.

[5] J. Candamo, M. Shreve, D.B. Goldgof, D.B. Sapper, and R. Kasturi: "Understanding Transit Scenes: A Survey on Human Behavior-Recognition Algorithms", IEEE Transactions on Intelligent Transportation Systems, vol. 11, no. 1, pp. 206–224, 2010.

[6] N. Buch, S. A. Velastin, and J. Orwell, "A review of computer vision techniques for the analysis of urban traffic", IEEE Trans. Intell. Transportation Syst., vol. 12, no. 3, pp. 920_39, Jun. 2011.

[7] R. Liang, Y. Zhu, and H. Wang, "Counting crowd flow based on feature points." Neurocomputing, vol. 133, pp. 377_84, Jun. 2014.

[8] B. Solmaz, B. E. Moore and M. Shah: "Identifying Behaviors in Crowd Scenes Using Stability Analysis for Dynamical Systems", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 10, pp. 2064–2070, October 2012.

[9] L. Kratz and K. Nishino: "Anomaly Detection in Extremely Crowded Scenes Using Spatio-Temporal Motion Pattern Models", Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2009.

[10] V. Mahadevan, W. Li, V. Bhalodia and N. Vasconcelos: "Anomaly Detection in Crowded Scenes", Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 1446–1453. Miami, FL. 2010.

[11] R. Mehran, A. Oyama, and M. Shah: "Abnormal Crowd Behavior Detection Using Social Force Model", Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 935–942, Miami, FL. 2009.

[12] O. P. Popoola and Kejun Wang: "Video Based Abnormal Human Behavior Recognition—A Review", IEEE Transactions on Systems, Man, and Cybernetics—part C: Applications and Reviews, vol. 42, no. 6, pp. 865–878, November 2012.

[13] J.Y. Choi, J.W. Choi and Y.K. Yang: "Improved tracking of multiple vehicles using invariant feature-based matching", Pattern Recognition and Machine Intelligence 4815, pp. 649–656, 2007.

[14] M. Balcells, D. DeMenthon and D. Doermann: "An appearance-based approach for consistent labeling of humans and objects in video", Pattern and Application, pp. 373–385, 2005.

[15] S. Calderara, R. Cucchiara and A. Prati: "Bayesian-competitive consistent labeling for people surveillance", IEEE Transactions on PAMI vol. 30, no. 2, pp. 354–360, 2008.

[16] W. Hu, M. Hu, X. Zhou, T. Tan, J. Lou and S. Maybank: "Principal axis-based correspondence between multiple cameras for people tracking", IEEE Transactions on PAMI vol. 28, no. pp. 663–671, 2006.

[17] F. Chang, G. Zhang , X. Wang, and Z. Chen: "PTZ Camera Target Tracking in Large Complex Scenes", in Proc. of the 8th World Congress on Intelligent Control and Automation, Jinan, China, July 2010.

[18] A. Gilbert & R. Bowden: "Incremental, scalable tracking of objects inter camera". Computer Vision and Image Understanding CVIU, vol. 111, no. 1, Springer, pp. 43–58, July 2008.

[19] S.N. Lim, L. Davis, and A. Elgammal: "Scalable image-based multicamera visual surveillance system", Proceedings. *IEEE* Conference on Advanced Video and Signal Based Surveillance, 2003, pp. 205–212, July 2003.

[20] J. Park, P. C. Bhat, and A. C. Kak: "A look-up table based approach for solving the camera selection problem in large camera networks", in Working Notes of the International Workshop on Distributed Smart Cameras (DSC 2006*)*, B. Rinner and W. Wolf, Eds., Boulder, CO, pp. 72–76, Oct 2006.

[21] J. Liu, S. Sridharan, C. Fookes and T. Wark: "Optimal Camera Planning Under Versatile User Constraints in Multi-Camera Image Processing Systems", IEEE Transactions on Image Processing, vol. 23, no. 1, January 2014.

[22] H.C. Shih, H.W. Lee, C.L. Huang and Y.C. Liu: "Collaborative Real-Time Scheduling for Multiple Objects Tracking in PTZ Camera Network", IEEE International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2012) November 4-7, 2012.

[23] F. Z. Qureshi and D. Terzopoulos: "Smart camera networks in virtual reality", Proceedings of the *IEEE* (Special Issue on Smart Cameras), vol. 96, no. 10, pp. 1640–1656, October 2008.

[24] F. Z. Qureshi and D. Terzopoulos: "Virtual vision and smart cameras", in Proc. of the First ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC07), Vienna, Austria, pp. 87–94, Sep. 2007.

[25] F. Z. Qureshi and D. Terzopoulos: "Surveillance camera scheduling: A virtual vision approach", ACM Multimedia Systems Journal, vol. 12, pp. 269–283, December 2006.

[26] G. Farnebäck: "Two-Frame Motion Estimation Based on Polynomial Expansion", SCIA13, Lecture Notes in Computer Science, vol. 2749, 363–370, 2003.

[27] M. Parimala, D. Lopez and N.C. Senthilkumar: "A Survey on Density Based Clustering Algorithms for Mining Large Spatial Databases", International Journal of Advanced Science and Technology, vol. 31, June, 2011.

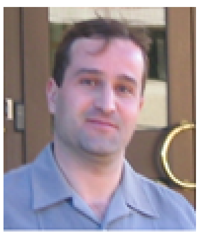
[28] W. He & Z. Liu: "Motion Pattern Analysis in Crowded Scenes by Using Density Based Clustering", 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2012), pp. 1855–1858. Sichuan 2012.

[29] F. Santoro, S. Pedro, Z.H. Tan and T.B. Moeslund: "Crowd Analysis by Using Optical Flow and Density Based Clustering", 18th European Signal Processing Conference (EUSIPCO), Denmark, pp. 269–273, August 2010.

[30] C. E. Shannon and W. Weaver: "The Mathematical Theory of Communication", University of Illinois Press, 1949.

[31] Transportation Research Board, "Highway Capacity Manual 2010". National Research Council, Washington, D.C, 2010.

[32] F. Z. Qureshi and D. Terzopoulos: "Distributed Coalition Formation in Visual Sensor Networks: A Virtual Vision Approach", Proceedings of the 3rd IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS'07), Santa Fe, NM, June, 2007.



Amar El Maadi received the Engineer degree in electrical systems from the University of Boumerdes, Algeria, in 1995, and the M.S. degree from the Université Laval, Québec-Canada, in 2006. He is currently working toward the Ph.D. degree at the Department of Control, Polytechnic Military Institute. His current research interests include patterns recognition and motion tracking. His publications include a number of conference and journal papers in Infrared thermography, computer vision, and signal processing.

Mohand Said Djouadi received the Engineer degree in electronics systems from the Polytechnic Military Institute, Algeria, in 1992, and the M.S. and Ph.D. degrees from the Polytechnic National Institute, Algeria, in 1998 and 2005, respectively. Currently, he is the head of the Department of Control at the Polytechnic Military Institute. His current research interests include, vision, robotics, sensor management and targets tracking. He has authored or coauthored many publications in the field of image and signal processing and robotics.

AUTHORS' ADDRESSES

Prof. Amar El Maadi, M.Sc.
Mohand S. Djouadi, Ph.D.
Department of Control,
Ecole Militaire Polytechnique,
BP: 17, DZ-35320 Bordj El Bahri, Algiers, Algeria
email: elmaadiamar@gmail.com, msdjouadi@gmail.com

Received: 2014-05-26
Accepted: 2015-08-26