Predictive and Probabilistic Tracking to Detect Stopped Vehicles

Rudy Melli, Andrea Prati, Rita Cucchiara
D.I.I. - University of Modena and Reggio Emilia
Via Vignolese, 905/b
I-41100 Modena, Italy

Lieven de Cock
Traficon N.V.
Meensesteenweg 449/2
B-8501 Bissegem, Belgium

Abstract

Many techniques and models have been proposed for vehicles surveillance in highways. In the past, tracking algorithms based on Kalman filter have been largely used for their efficiency in the prediction and low computational cost. However, predictive filters can not solve long-lasting occlusions. In this paper, we propose a new mixed predictive and probabilistic tracking that exploits the advantages of predictive filters for moving vehicles and adopts probabilistic and appearance-based tracking for stopped vehicles. The proposed tracking is part of a complete video surveillance system, oriented to control tunnels and highways from cluttered views, that is implemented in an embedded DSP platform and provides background suppression, a novel shadow detection algorithm, tracking, and scene recognition module. The experimental results are obtained over several hours of videos acquired in pre-existing platforms of CCTV surveillance systems.

1. Introduction and related works

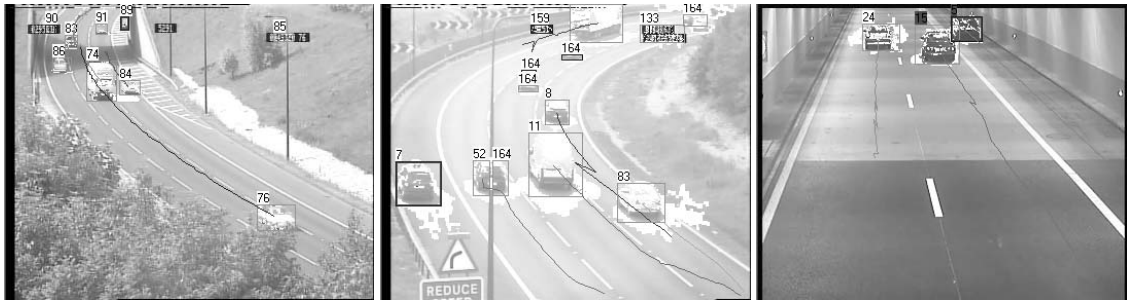
Tracking objects for video surveillance systems can be considered a well assessed task, provided that the working conditions are optimal. Unfortunately, in real situations, optimal conditions are quite rare, and many additional problems arise. Our target application is traffic monitoring in outside cluttered environments, and the main purpose is the detection of *vehicles that stop in forbidden areas*. This is of particular interest in tunnels, where unauthorized stops can be of great danger for the driver's safety.

Existing commercial ITS (Intelligent Transportation Systems) solutions rarely exploit temporal information of the scene: objects are detected by simple motion detection techniques, such as inductive loop emulation. However, these approaches can not track the whole objects. Tracking of moving objects is fundamental to keep history and identity of objects when they stop, and, thus, to understand if the detected stopped blob belongs to a real object (being tracked in the recent past) or not.

Suitable tracking algorithms should exhibit robustness to shape changes (often due to perspective changes), to appearance changes (due to illumination variations, auto-iris, sun reflections and shadows) and to occlusions (many frequent in real high traffic scenes). Problem of occlusion management is particularly critical for stopped vehicles (see for example Fig. 1(a) and Fig. 1(b)). In fact, the typical installations orient the point of view to the main lane and the emergency lane is generally occluded by other vehicles.

These requirements should be met together with the constraint of high computational efficiency. Working embedded systems with DSP technology cannot support the same computational load as general-purpose workstations. Indeed, most of the recent research solutions providing objects tracking are normally presented as expensive software applications running on high performance PCs, not affordable for real large-scale applications with forest of cameras distributed over many kilometers of highways.

In this paper, we present and discuss a novel approach for detecting and tracking of dangerous stopped vehicles in highways with cluttered conditions. Key aspects of the proposal are the adoption of a selective background model and a shadow removal algorithm in YCbCr space. The main novelty of the work is the robust and efficient technique of prediction and probabilistic vehicle tracking. Most of directional tracking approaches, generally based on motion prediction, such as Kalman filters, work on single object points (e.g., the centroid). They typically require limited computational power and are efficient enough to be used for vehicle tracking on the road, but they are not capable of managing long-lasting occlusions. Proposed extensions with heuristic assumptions cope with partial or short-lasting occlusions by accepting tracking inconsistency while the occlusion occurs, and matching object labels before and after occlusion. Indeed, they lose all the information when the objects are not visible. On the other hand, probabilistic and appearance based methods process all the object points, obtaining, in general, high reliability at the cost of a higher computational



(a) Occlusion due to trees (b) Occlusion due to poles (c) Tunnel example

Figure 1. Examples of the problems in real setups.

load. They are suitable for shape changes and occlusions (in fact, they are typically used for people tracking) since they maintain the history of previous object appearance by means of a probabilistic framework. In this work, we propose a mixed approach that exploits the speed of Kalman technique in not critical situations and improves the stopped vehicles detection with an appearance-based model.

Tracking methods can be classified according with the motion model adopted. In the case of *tracking of multiple rigid objects*, such as in ITS, the objects can be considered with rigid motion and their appearance mainly changes due to the perspective. The trajectory is easily predictable, like in the case of vehicles following the road lane's direction. Kalman filter prediction [8] (or modifications such as the Extended Kalman filter [3] or the CONDENSATION algorithm [4]) are resulted to be quite effective in these situations. Differently, other statistical approaches have been proposed, such as first-order statistics [11], mixture of Gaussians [12] or Markov Random Fields [7], but they are generally very time consuming. The most critical problems concern merging and splitting of detected blobs, also called explicit and implicit occlusions [5]. These problems are often solved by means of reasoning rules or graph matching approaches. In [2] a forward chaining production rule system is used, tailored to intersections. In [5] rules for checking temporal graph merging or splitting are defined. In [13] blob matching and occlusion rules are defined in a complex intersection monitoring system. Other works exploit specific features to improve tracking in occlusion handling: for example, [3] uses the object skeleton and [9] employs the corners with the generalized Hough transform.

In the case of *tracking of multiple non-rigid objects*, such as people tracking, in addition to merging and splitting, the tracking system must also take into account large occlusions and shape changes. In [11] a probabilistic approach has been proposed in vehicle and parking control as well as in people tracking. It can manage short-lasting occlusions. This proposal is very promising, but requires a heavy computational time proportional to the number of objects, so that is not affordable in highway where tens of objects must be tracked at the same time.

2. System architecture

The system is composed by three main modules: segmentation, tracking, and scene understanding (Fig. 2).

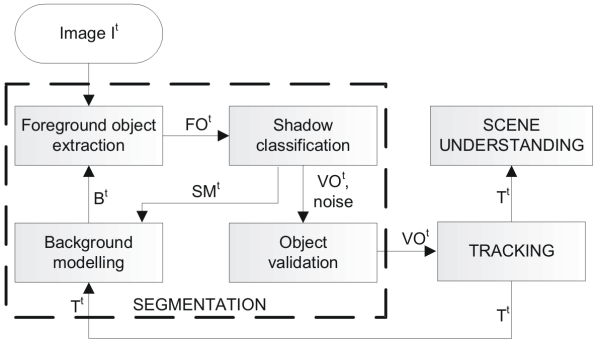


Figure 2. The architecture of the system.

The segmentation module aims at extracting the *visual objects*. The first step uses the background suppression by subtracting the current background model B^t from the current frame I^t. The points are extracted and grouped with a labelling process into a set FO^t of foreground objects at instant time t. This set contains both relevant objects and other outliers, such as shadows and noise.

To identify shadow points we used a deterministic approach, similar to that proposed in [10], but duly modified for computational reasons to work directly in the YCbCr space (instead of the HSV space adopted by [10]). A point p (resulted from the segmentation) is classified as shadow

	HSV		YCbCr	
	$\eta\%$	$\xi\%$	$\eta\%$	$\xi\%$
Highway1	64.73%	60.98%	62.49%	64.63%
Highway2	66.25%	78.44%	63.89%	70.39%

Table 1. Shadow detection comparison between HSV and YCbCr spaces

point if its value in the mask $SM^t(p)$ is 1:

$$SM^t(p) = \begin{cases} 1 & if \ \ \alpha \leq \frac{I^t_V(p)}{B^t_V(p)} \leq \beta \wedge \ \Psi \leq \tau_S \wedge \ \Phi \leq \tau_H \\ 0 & otherwise \end{cases} \tag{1}$$

where the subscript Y denotes the Y component of a vector in the YCbCr space, and $\alpha, \beta \in [0, 1]$. The values Ψ and Φ can be obtained from equations reported in [10] approximating the hue H with the angle between vectors formed by Cb and Cr and the saturation S with the respective modules difference:

$$\Psi = \left| \sqrt{(I^t_{Cr})^2 + (I^t_{Cb})^2} - \sqrt{(B^t_{Cr})^2 + (B^t_{Cb})^2} \right|$$
$$\Phi = \frac{\left(I^t_{Cb} B^t_{Cb} + I^t_{Cr} B^t_{Cr}\right)^2}{\left((I^t_{Cb})^2 + (I^t_{Cr})^2\right)\left((B^t_{Cb})^2 + (B^t_{Cr})^2\right)}. \tag{2}$$

We have compared the results achieved with HSV with respect to our proposal, computing metrics of *shadow detection rate* η or *shadow discrimination rate* ξ, proposed in [10]. These metrics allow to evaluate the capacity of detecting shadow points and that of not classifying foreground point as shadow, respectively. A comparison between the HSV-based approach and our proposal on ground-truthed videos is reported in Table 1.

Small objects extracted by the segmentation module are considered as noise and discarded; same rules based on object dimension and taking into account perspective distortion are adopted.

The set VO^t of visual objects obtained after the size-based validation is processed by the tracking module that computes for each frame t a set of tracks $T^t = \{T^t_1, ..., T^t_m\}$ and assigns to each track T^t_i a status label: *moving*, *stopped*, *apparent*, *new*, or *undetected*. An object is classified as *stopped* when it is detected as still in the current frame, but it was moving in at least a certain number of previous frames. The case of *apparent* object occurs whenever a still object starts. It is well known that this generates two objects in background suppression methods: the first corresponding to the real moving object, the second corresponding to the difference between the background image (where the object is still) and the current image (where the object has moved). This case can be discriminated from the case of stopped real object by exploiting the tracking.

The knowledge about VO and their status is exploited by a selective background model. To this aim we adopt the model proposed in [1] but with some changes. In particular the above mentioned track status is used to selectively update the background model and to exclude stopped vehicles from the updating.

Scene understanding is a high level module. It receives the information from the tracking and switches on an alert when a stopped vehicles is detected.

3. Tracking

The vehicle tracking is performed by a new mixed tracking approach based on two different algorithms.

The basic tracking module is a predictive Kalman filter, wherewith the correspondences between Visual Objects and tracks are estimated, together with the computation of the motion vector $\vec{e}$ of the tracks; in particular, the Kalman filter tracks the centroid position. Using the well-known notation of the discrete Kalman filter [6], the state x_k and the measurement z_k adopted are represented by the vectors $x_k = (p_k, s_k, a_k)^T$ and $z_k = (p_{k-2}, p_{k-1}, p_k)^T$, where p is the position, s is the speed, and a is the acceleration of the centroid.

The tracking process starts with the association between the VO^t and T^{t-1} sets. To this aim a Boolean *correspondence matrix* is created. Supposing each VO and each tracks T are provided together with its bounding box BB, its color image M, and its centroid c, a visual object and a track are marked as correspondent if the *Bounding Box Distance* (BBd) defined in Eq. 3 is low enough.

$$BBd(VO_j, T_k) = \\ = \min_{\substack{x_k \in BB_k, \\ y_j \in BB_j}} \left(\min\left\{\|c_j, x_k + \vec{e}_k\|, \|c_k + \vec{e}_k, y_j\|\right\}\right) \tag{3}$$

In case of one-to-one $VO - T$ correspondence the Kalman filter is used. The track is labeled as *moving* or *stopped* depending on its motion. It is labeled as *new* in the case of a one-to-zero correspondence, while the *undetected* state is set for zero-to-one correspondences. It is labeled as *apparent* if it was *new* and remains without motion until a timeout. Instead, if a blob VO^t_j is contended by two or more old tracks T^{t-1}_k ($\in T^{t-1}$), an occlusion is going to take place. In particular, if a vehicle is classified as stopped, this group of occluded blobs starts to be processed with a probabilistic approach (defined later) until the end of occlusion. Eventually, in order to cope with the *n-to-n* correspondences between VOs and tracks, we define the concept of the *Macro-Object* (MO) as the union of VOs associated with the same track T_k. After that, the probabilistic algorithm is applied to the MOs in the same way of the VO.

The probabilistic approach is based on the computation of an *appearance image AI* (or *temporal template*) and a *probability mask PM* of the track. AI is the estimated aspect of the track and it is obtained with a temporal integra-

tion of the color images of the VOs, while the probability mask associates to each point a real value (between 0 and 1) that indicates its probability to belong to the track. Eventually, a *probability of non occlusion* PNO_k is associated with the whole track.

For each track T_i the estimated position obtained by a constant velocity assumption is refined with the displacement $\vec{\delta} = (\delta_x, \delta_y)$ that maximizes a fitting function P_{FIT}. This process is iterated for all the tracks associated with a MO, with an order proportional to their probabilities of non-occlusion PNO and considering only the pixel not jet assigned. The P_{FIT} is:

$$P_{FIT}(T_k, \vec{\delta}_{BF}) = Likelihood \cdot Confidence$$

$$Likelihood = \frac{\sum_{x \in MO} P_{APP}\left(I(x - \vec{\delta}_{BF}), AI_k(x)\right) \cdot PM_k(x)}{\sum_{x \in MO} PM_k(x)}$$

$$Confidence = \frac{\sum_{x \in MO} PM_k(x)}{\sum_{x \in T_k} PM_k(x)} \quad (4)$$

The first term is a measure of how similar are the corresponding pixels of the MO and the track; the second term is the percentage of track points, weighted with their probability, that are visible on the current frame and belong to the MO. Accordingly, when the product of $Likelihood$ and $Confidence$ is low, the track is considered totally occluded (and $\vec{\delta}_{BF}$ is not used).

$P_{APP}\left(I(x - \vec{\delta}_{BF}), AI_k(x)\right)$ measures the correspondence between the actual YCbCr color of the point in MO and the appearance model of the track. As in [11], we use a spherical Gaussian to approximate the pixel color distribution around the mean stored by the model, that is:

$$P_{APP}(\mathbf{s}, \mathbf{t}) = \left(2\pi\sigma^2\right)^{-\frac{3}{2}} e^{-\frac{\|\mathbf{s}-\mathbf{t}\|^2}{2\sigma^2}} \quad (5)$$

where the norm used is the Euclidean distance in YCbCr space.

Once the tracks have been aligned, all the MO points must be assigned to a track. The points of the MO contended by the different tracks are assigned exploiting a Bayesian function:

$$P(T_k|x) = \frac{P(x|T_k)\, P(T_k)}{\sum_{i=1}^{m} P(x|T_i)\, P(T_i)} \quad (6)$$

The conditional probability is the product of two terms: $P(x|T_k) = P_{APP}(x) PM_k(x)$. This probability takes into account the difference between the colors of the actual pixel and the track appearance one (Eq. 6 computed on a single pixel x), weighted by the probability that the point belongs to the track. In order to cope with track mutual occlusions, the $P(T_k)$ is suitably modelled as the *a-priori probability* of seeing T_k, defined as a probability of non occlusion. Each point will be assigned to the track that maximizes $P(T_k|x)$ and the set of point assigned to the track T_k is named A_k.

An occluded track is detected since the $Confidence$ value of Eq. 4 goes lower than a threshold. In this case the track model should be "frozen" since the memory of the object's appearance should be preserved. Nevertheless, if the $Confidence$ decreases due to a sudden shape motion (apparent occlusion), not updating the track would lead to an error. For discriminating between actual and apparent occlusions, the non-visible points $NV_K^T = \{T^t - A_k\}$ are grouped in regions (by a labelling process and discarding small regions), classified in apparent R_{AO} and real occlusions R_{RO}. This classification is then exploited to selectively update the track model.

As the final step, the masks are updated with adaptive functions. In particular, $\forall x \in T^t$:

$$PM^t(x) = \begin{cases} \lambda PM^{t-1}(x) + (1-\lambda) & x \in A_k \\ PM^{t-1}(x) & x \in R_{RO} \\ \lambda PM^{t-1}(x) & otherwise \end{cases} \quad (7)$$

$$AI^t(x) = \begin{cases} \lambda AI^{t-1}(x) + (1-\lambda) I^t(x) & x \in A_k \\ AI^{t-1}(x) & otherwise \end{cases} \quad (8)$$

The initial value used for $PM^t(x)$ is 0.6, because, since the vehicles remain in the scene only few frame, the initial probability must not be too low. Moreover, for the same reason, it is necessary to have a quite fast update of the model (λ=0.9). Defining PO_{ik}^t as the probability that track T_i occludes T_k, the non-occlusion probability is computed as a value proportional to the number a_{ik} of shared points assigned to T_i and not to T_k.

In particular, $PNO^t(T_k) = 1 - \max_{i=1,\ldots,m}(PO_{ik}^t)$ and defining $\beta_{ik} = \frac{a_{ik}+a_{ki}}{\|A_i\|}$, the update model of PO_{ik}^t is:

$$PO_{ik}^t = \begin{cases} 0 & \beta_{ik} < \vartheta_o \\ (1-\beta_{ik})\, PO_{ik}^{t-1} & a_{ik} = 0 \\ (1-\beta_{ik})\, PO_{ik}^{t-1} + \beta_{ik} e^{-\frac{a_{ki}}{a_{ik}}} & a_{ik} \neq 0 \end{cases} \quad (9)$$

where ϑ_o is a suitable threshold.

Finally, the motion centroid position of the track retrieved from the previous calculation is passed to the Kalman filter to estimate the vector $\vec{e}_k (= sp_k)$.

4. Experimental results

This work has been obtained with the synergy between the research activity at university and the precompetitive prototypal works at Traficon N.V., Belgium (`http://www.traficon.com`) for addressing

most of these real problems, related to illumination conditions, object's density and speed, reliable camera position and parameters, and available hardware. Therefore, we have the availability of tens of existing CCTV source sites to experiments our system in many different conditions. In table 2 some examples of videos are reported with clips selected for the presence of stopped objects. Stopped vehicles are not rare events in a 24h hours traffic surveillance and the system must exhibit a very high recall (on 100% of the events). The most frequently error is when a stop track is lost and detected as a new track; it can be classified as a apparent object, because in the middle of the scene and without track history, then put in the background model and lost.

The proposed process of object segmentation is quite robust in many different illuminance conditions. Fig. 1 shows vehicles segmentation and tracking in different real setups; the shadows are indicated in white. Fig. 1(a) shows how the shadow detection works fine by identifying the shadow of the track and without merging the tracks 74 and 84. Fig. 1(b) highlights the problems of Kalman; due to the occlusion, track 52 is not identify completely, but divided in two tracks (one new). Finally, Fig. 1(c) shows an example of tunnel where stops are very dangerous and occlusions due to the position of the camera, and thus to the high perspective distortion, can be very frequent.

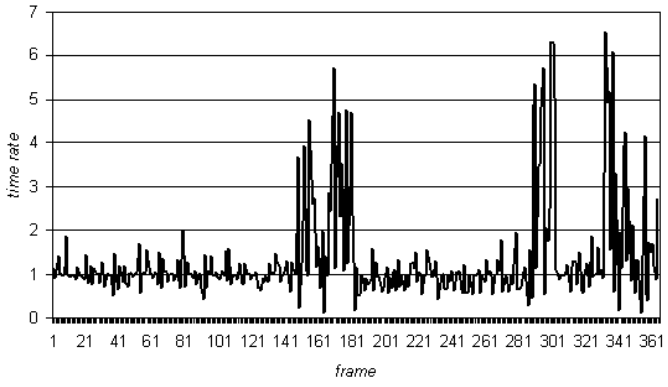


Figure 3. Time comparison between Kalman tracking and mixed approach.

The Kalman-based vehicle tracking is largely used for its good prediction and acceptable computational power, in general the results can be easily improved by adding constraints such as predictive trajectory and directional search, and adjusting the parameters in dependance of the installation. Probabilistic tracking is absolutely very robust and suitable for occlusions. Although its efficacy, it cannot be adopted for all the moving objects of the scene: tests demonstrate that the computational time becomes unbearable when the number of tracks is more than 4 or 5 or of equivalent size. A mixed approach can skirt the problem limiting the use of probabilistic model only in given cases of interest, i.e. stopped vehicles. Fig. 3 reports the ratio between the tracking time with a pure Kalman filter and with our mixed approach. The ratio is an average 1 (meaning that in both tests only Kalman is enabled), but, in the 3 different peaks (Fig. 3, intervals $150-180$, $290-305$, $330-360$), the increase of time due to the probability tracking (caused by occlusions) is evident. Thus, this hybrid approach allows better performance in occlusion management and it remains feasible for implementation on (limited) DSP resources.

Several tests on various videos obtained from real installations with different occlusions have been carried out, as described in Table 2. From it, it is evident that Kalman-based tracking tends to detect many false positives due to the occlusions. Instead, hybrid approach is able to detect most of the stopped vehicles: missed detections are due to very large occlusions and to frequent auto-iris releases.

In the example of Fig. 4, a limited sequence is reported where a car is occluded by another vehicle. We have reported the appearance models ((c) and (e)) and the probability masks ((b) and (d)) of both vehicles in the occlusion sequence ((a)) resolved with probabilistic approach. In Fig. 4 (e) is drawn the graph of the trend of $Likelihood$ and $Confidence$ of both vehicles. During the occlusion the $Likelihood$ of stopped car ($track$ 158) is approximately constant because the points assigned to this track are very similar to its appearance model, but the $Confidence$ decreases because few points have been assigned to this track. For the other track (188), the values of $Likelihood$ and $Confidence$ maintain an approximately constant trend. This is a clear sign of occlusion and the occluded object is with evidence the stopped track. Indeed, during the occlusion, the probability masks of the occluded vehicles does not change (Fig. 4(b)); instead, the other one becomes every frame more white (Fig. 4(d)), then probability increases, that means that many points have been assigned to this track and the vehicle is in the front layer.

5. Conclusions

We have proposed a mixed tracking approach and we have shown how appearance models and probability masks can be used in conjunction with Kalman filter-based tracking to solve occlusions on real-time vehicle tracking focused on stop detection. Thanks to the synergy of both algorithms the computational cost of the resulting tracking does not invalidates the work of the system, allowing a better and more reliable tracking in case of long-lasting occlusions. The proposed vehicles tracking contribution is pre-

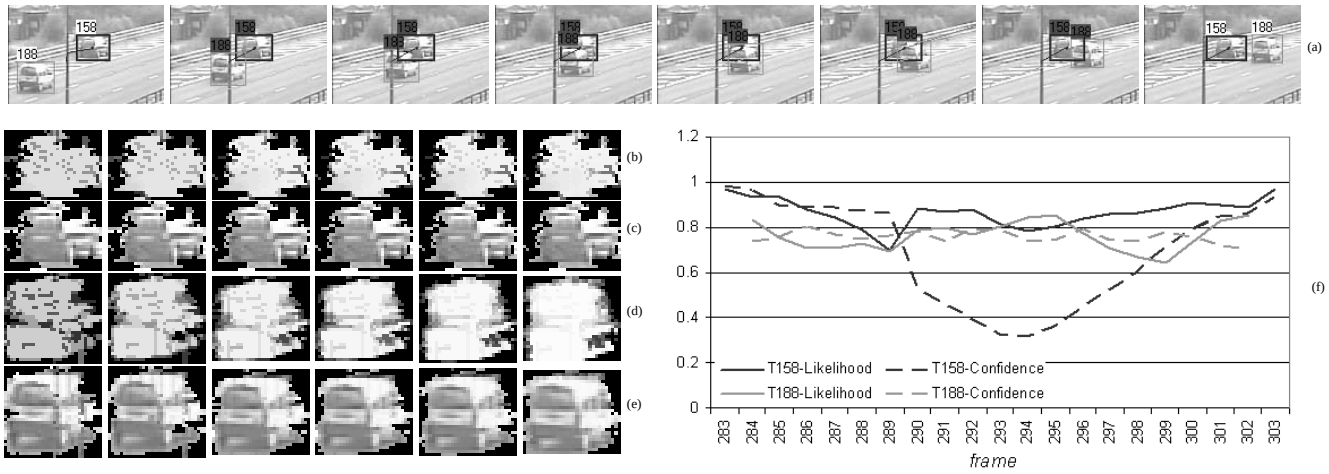


Figure 4. Example of occlusion tracked with probabilistic tracking.

Camera	Description	frames, time	vehicles	stopped frames	stops	Detected (Kalman)	Detected (Hybrid)
Manch. Cam5551	Traffic cong., 2 ways	6621, 4'24"	439	4550	3	7	2
Manch. Cam5592	Occlusion (bridge), 2 ways	1377, 0'55"	131	576	1	3	1
Manch. Cam5251	Occlusion (poles), 2 ways	4581, 3'03"	149	2981	3	4	3
Manch. Cam5291	Occlusion (trees), 1 way	5457, 3'38"	140	4175	3	6	3
Luxem. Cam 2	Tunnel - low ill.	25594, 17'03"	37	17350	12	10	10
Luxem. Cam a	Tunnel - track occl.,1 way	16450, 10'58"	65	6033	4	6	4

Table 2. Video examples, of real installations, with stopped vehicles

sented within the framework of an existent embedded real application system integrated on a DSP device with a new approach for shadow detection on YCbCr color space.

References

[1] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati. Detecting moving objects, ghosts and shadows in video streams. *IEEE Trans. on PAMI*, 25(10):1337–1342, Oct. 2003.

[2] R. Cucchiara, M. Piccardi, and P. Mello. Image analysis and rule-based reasoning for a traffic monitoring system. *IEEE Trans. on Intelligent Transportation Systems*, 1(2):119–130, June 2000.

[3] G. Foresti. Object recognition and tracking for remote video surveillance. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(7):1045–1062, Oct. 1999.

[4] M. Isard and A. Blake. A smoothing filter for condensation. In *Proc. of European Conference on Computer Vision*, volume 1, pages 767–781, 1998.

[5] Y.-K. Jung and H. Y.-S. Traffic parameter extraction using video-based vehicle tracking. In *Proc. of IEEE Int'l Conference on Intelligent Transportation Systems*, pages 764–769, 1999.

[6] R. E. Kalman. A new approach to linear filtering and prediction problems. In *Transaction of the ASME-Journal of Basic Engineering*, pages 35–45, Aug. 1960.

[7] S. Kamijo, Y. Matsushita, K. Ikeuchi, and M. Sakauchi. Traffic monitoring and accident detection at intersections. *IEEE Trans. on Intelligent Transportation Systems*, 1(2):108–118, June 2000.

[8] H. Nguyen and A. Smeulders. Template tracking using color invariant pixel features. In *Proc. of IEEE Int'l Conference on Image Processing*, volume I, pages I–569–I–572, 2002.

[9] F. Oberti, M. Calcagno, S. Zara, and C. Regazzoni. Robust tracking of humans and vehicles in cluttered scenes with occlusions. In *Proc. of IEEE Int'l Conference on Image Processing*, volume 3, pages 629–632, June 2002.

[10] A. Prati, I. Mikic, M. Trivedi, and R. Cucchiara. Detecting moving shadows: Algorithms and evaluation. *IEEE Trans. on PAMI*, 25(7):918–923, July 2003.

[11] A. Senior. Tracking people with probabilistic appearance models. In *Proc. of Int'l Workshop on Performance Evaluation of Tracking and Surveillance (PETS) systems*, pages 48–55, 2002.

[12] C. Stauffer and W. Grimson. Learning pattern of activity using real-time tracking. *IEEE Trans. on PAMI*, 22(8):747–757, Aug. 2000.

[13] H. Veeraraghavan, O. Masoud, and N. Papanikolopoulos. Computer vision algorithms for intersection monitoring. *IEEE Trans. on Intelligent Transportation Systems*, 4(2):78–89, June 2003.