A Fast Recursive Approach to Autonomous Detection, Identification and Tracking of Multiple Objects in Video Streams under

Pouria Sadeghi-Tehran[1], Plamen Angelov[1], and Ramin Ramezani[2]

[1] Department of Communication Systems, Infolab21,
Lancaster University Lancaster, LA1 4WA, United Kingdom
p.sadeghi-tehran@lancaster.ac.uk, p.angelov@lancs.ac.uk
[2] Department of Computing, Imperial College London, United Kingdom
raminr@doc.ic.ac.uk

Abstract. Real-time processing the information coming form video, infra-red or electro-optical sources is a challenging task due the uncertainties such as noise and clutter, but also due to the large dimensionalities of the problem and the demand for fast and efficient algorithms. This paper details an approach for automatic detection, single and multiple objects identification and tracking in video streams with applications to surveillance, security and autonomous systems. It is based on a method that provides recursive density estimation *(RDE)* using a Cauchy type of kernel. The main advantage of the RDE approach as compared to other traditional methods (e.g. KDE) is the low computational and memory storage cost since it works on a frame-by-frame basis; the lack of thresholds, and applicability to multiple objects identification and tracking. A robust to noise and clutter technique based on spatial density is also proposed to autonomously identify the targets location in the frame.

1 Introduction

Uncertainties are inherently related to video streams and can broadly be categorised as;

i) noise (rather probabilistic disturbances and errors);
ii) clutter (correctly identifying objects that are however of no interest to the observer – e.g. not a target that we want to track etc.).

Processing in real-time information that is coming form image, infra-red (IR) or electro-optical (EO) sources is a challenging task due to these uncertainties, but also due to the large dimensionalities of the problem (the resolution nowadays allow having millions of pixels and the rates of collecting information in order of dozens or more frames per second). At the same time the demand from applications that are related to surveillance, security and autonomous systems require fast and efficient algorithms. Recently, the use of security and surveillance systems is the centre of attention due to growing insecurity and terrorism activities

E. Hüllermeier, R. Kruse, and F. Hoffmann (Eds.): IPMU 2010, Part II, CCIS 81, pp. 30–43, 2010.

around the world. A pressing demand is the problem of automating the video analytical processes which require short processing time and low memory and storage requirement to enable real-time autonomous applications.

Traditional visual surveillance systems are not very efficient since they require a large amount of computer storage to archive video streams for further batch mode processing [1-3, 16]. They also often rely on manual (as opposed to automatic) and off-line target/object identification. One of the most widely used approaches for novelty detection is based on so called background subtraction [4-7]. This approach is based on building a representation of the scene background and compares new frames with this representation to detect unusual motions [4] . Instead of using window of consecutive frames to build background and keep them in the memory for off-line processing [4, 5], we propose a fully autonomous analysis on a per frame basis which is using recursive calculations and removes the need of computer storage to archive video frames. Additionally, the introduced approach is threshold-independent and minimises the processing time by discarding the unnecessary data. The main idea of the proposed approach is to approximate the probability density function (pdf) using a Cauchy type of kernel (as opposed to Gaussian one used in KDE technique), and then in order to update this estimation we apply a recursive expression using the colour intensity of each pixel. In this manner, only the accumulated information which represents the colour intensity of each pixel is stored in the memory and there is no need to keep huge volumes of data in the memory. As a result, the proposed technique is considerably (in an order of magnitude) faster and more computationally efficient.

The second innovation that is introduced in this paper is the automatic single and multiple object(s) identification in the frame. For the newly proposed multi-object detection we use a novel clustering technique to group the foreground pixels which represents objects/targets and distinguish them from the noise (due to luminance variation) and clutter. The proposed approach can be extended for tracking objects using Kalman Filter (KF) or evolving Takagi-Sugeno fuzzy model [8] , and landmark detection used in robotics [9, 10].

The remainder of the paper is organised as follows. In section two, the RDE novelty detection in video streams method is introduced. First, the widely used method KDE is explained and then its recursive version, RDE is introduced. The problem of single and multi-objects tracking and the mechanism for approaching this problem is explained in section 3. Section 4 represents the tracking technique based on eTS fuzzy system. Section 5 displays the experimental results. At the end, section 6 provides conclusion and discussion.

2 Novelty Detection in Video Streams through Recursive Densirty Estimation

2.1 Background Subtraction

One of the most popular and widely used methods for visual novelty detection in video stream is background subtraction method (BS) [4, 7]. Background subtraction is a method used to detect unusual motion in the scene by comparing each new frame to a model of the scene background. It is based on statistical modelling the background of the scene to achieve a high sensitivity to detect a moving object and robust to the noise. Robustness is required to distinguish fluctuations in the statistical characteristic due to non-rigid objects and noise, such as tree branches and bushes movements, luminance change, etc.

In [6] the absolute difference between every two frames is calculated and a threshold is used for decision making and model a foreground. As result, this method has low robustness to noise (e.g. luminance, variations, movement of tree branches, etc.) and clutter.

In order to cope with this problem a window of frames with length N (usually $N > 10$ is defined and analyzed in an off-line mode. Each pixel in the video frame is modelled separately as a random variable in a particular feature space and estimates its probability density function (pdf) across the window of Nframes [4, 5] (Fig. 1). The pdf is usually modelled as Gaussian. A more advanced approach is based on mixture of Gaussian (rather than a simple Gaussian) which is more realistic [11]. A drawback of this method is also using a threshold to selecting the proper distribution as a background model.

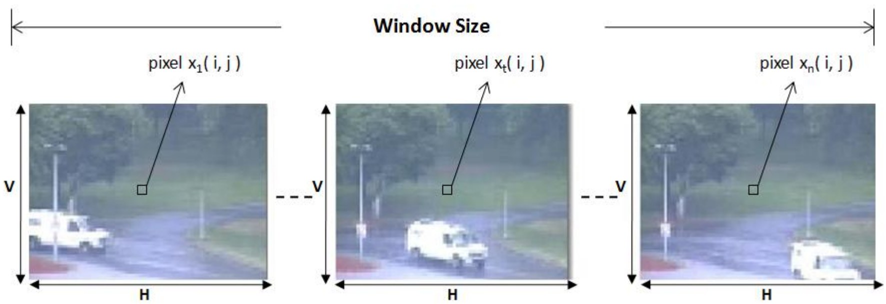

Fig. 1. Window of N frames used in KDE approach, H denotes the number of pixel in the horizontal and V – the number of pixels in the vertical

2.2 Kernel Density Estimation

Some of the most common techniques for modelling the background in video stream processing are non-parametric techniques such as the well known Kernel Density Estimation (KDE) [4]. Different types of kernels can be used to represent the pdf of each pixel of the frame each one having different properties. Typically, the Gaussian kernel is used for its continuity, differentiability, and locality properties [4].

$$p(z_t^{ij}) = \frac{1}{N}\sum_{r=1}^{N}\prod_{l=1}^{n} k_\sigma(z_{tl}^{ij} - z_{rl}^{ij}) \tag{1}$$

where k_σ denotes the kernel function (sometimes called a "window" function) with bandwidth (scale) σ; n denotes the colour channel (R,G,B or H,S,V) or, more generally, the number of input features; $z^{ij} = \left[z_1^{ij}, z_2^{ij}, \ldots, z_t^{ij}, \ldots, z_N^{ij}\right]^T; z \in R^n$ denotes the colour intensity values of N consecutive frames of a video stream that have a specific $(i,j)^{th}$ position in each frame (Fig. 1); $i=[1,H]$; $j=[1,V]$. If Gaussian is chosen to be a kernel function k_σ, then the colour intensity can be estimated as:

$$p(z_t^{ij}) = \frac{1}{N}\sum_{r=1}^{N}\prod_{l=1}^{n}\frac{1}{\sqrt{2\pi\sigma_l^2}}e^{-\frac{1}{2}\frac{(z_{tl}^{ij}-z_{rl}^{ij})^2}{\sigma_l^2}} \tag{2}$$

This can be simplified as:

$$p(z_t^{ij}) = \frac{1}{N\sqrt{2\pi\sigma_l^2}}\sum_{r=1}^{N}e^{-\sum_{l=1}^{n}\frac{(z_{tl}^{ij}-z_{rl}^{ij})^2}{2\sigma_l^2}} \tag{3}$$

Once the pdf is estimated by calculating the kernel function, it should be classified as a background (BG) or foreground (FG) by comparing to the pre-defined threshold [4].

$$IF\ (\ p(z_t^{ij}) < threshold)\ THEN\ (\ z_t^{ij}\ is\ foreground)\ ELSE\ (\ z_t^{ij}\ is\ background) \tag{4}$$

Although non-parametric kernel density estimation is very accurate, it is computationally expensive and the significant disadvantage of this method is the need to use a threshold. A wrong choice of the value of the threshold may cause a low performance of the whole system in difference outdoor environment. Another major problem/difficulty is to define a proper bandwidth for the kernel function. Practically, since only a finite number of samples are used and the computation must be performed in real time, the choice of suitable bandwidth is essential. Too small value of the bandwidth may lead the density estimation to be over-sensitive, while a wide bandwidth may cause the density estimation to be over-smoothed.

2.3 The Concept of the Proposed RDE Approach

The main idea of the proposed RDE approach is to estimate the pdf of the colour intensity given by equation (1)-(3) using a Cauchy type kernel (instead of Gaussian kernel) and calculate it recursively [12]. Such a recursive technique removes the dependence of a threshold and parameters (such as bandwidth) and allows the image frame to be discarded once they have been processed and not to be kept in the memory. Instead, information concerning the colour intensity per pixel is accumulated and is being kept in the memory. In this way, the amount of information kept in the memory is significantly smaller than original KDE approach, namely *(n+1)*H*V* or *(n+1)* per pixel compare to KDE which needs *(n*N*H*V)* data stored in the memory.

The Gaussian kernel can be approximated by a Cauchy function since the Cauchy function has the same basic properties as the Gaussian [13]. *a)* It is monotonic; *b)* its maximum is unique and of value 1; *c)* it asymptotically tends to zero when the argument tends to plus or minus infinity.

In RDE approach with using Cauchy type function the density of a certain (ij^{th}) pixel is estimated based on the similarity to **all** previously image frames (unless some requirements impose this to be limited to a potentially large window, N) at the same ij^{th} position.

$$D(z_t^{ij}) = \frac{1}{1 + \sum_{l=1}^{N} \sum_{r=1}^{n} \frac{(z_{tl}^{ij} - z_{rl}^{ij})^2}{2\sigma_r^2}} \tag{5}$$

It is very important that the density, D can be calculated recursively as demonstrated for other types of problems in [12, 13]. In a vector form (for all pixels in the frame) we have:

$$D(z_{tl}) = \frac{t-1}{(t-1)(z_t^T z_t + 1) - 2c_t + b_t} \tag{6}$$

Value c_t can be calculated from the current image frame only:

$$c_t = z_t^T d_t \tag{7}$$

where d_t is calculated recursively:

$$d_t = d_{(t-1)} + z_{(t-1)}; \; d_1 = 0 \tag{8}$$

The value b_t is also accumulated during the processing of the frames one by one as given by the following recursive expression:

$$b_t = b_{t-1} + \|z_{t-1}\|^2; \; b_1 = 0 \tag{9}$$

As mentioned earlier, to identify a foreground (novelty) the density of each ij^{th} pixel of the image frame is compared to pixels at the same ij^{th} position in **all** previous frames. In this way, the expression (10) should be applied for each pixel, (Fig. 2). It should be highlighted that in RDE approach there is no need to pre-define any threshold since we estimate the statistical properties of the density:

$$IF\left(D(z_t^{ij}) < \min_{l=1}^{t} D(z_l^{ij} - std(D(z_l^{ij}))\right) THEN\ (\ z_l^{ij}\ is\ FG)\ ELSE\ (\ z_l^{ij}\ is\ BG)\ \ i=[1,H];j=[1,V] \tag{10}$$

where $std\left(D\left(z_l^{ij}\right)\right)$ is the standard deviation of the densities of image frames seen so far.

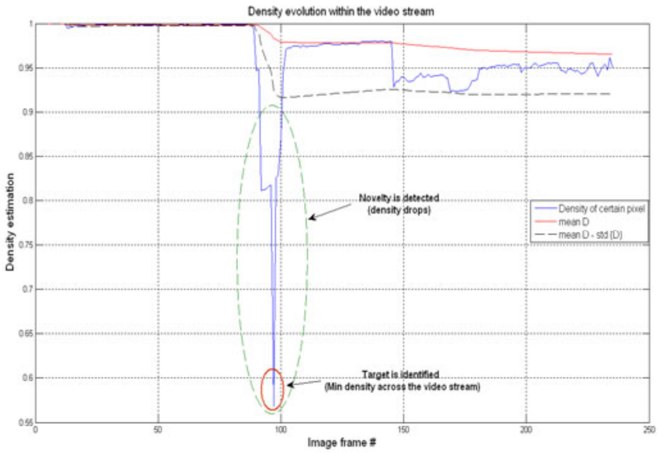

Fig. 2. The frames for which the value of the density drops below the value of meanD-std(D) are denoted by red circle and a novelty is detected there

3 Single/Multi Object(S) Identification Using RDE

3.1 Single Object Identification

After applying condition (11) to each pixel and detecting a novelty at a pixel level, the standard way to identify the object for tracking purposes is to find the spatial **mean value** of all pixels that have been identified to be background [5]. The drawback of this technique is the influence of the noise caused by change of illumination, move of tree branches and bushes, clutter, etc. This may lead to locating the object in a wrong position which might be misleading for the tracking. An alternative that is also often used for target tracking in video streams is the manual identification of the target/object which is obviously an off-line process [5]. In this paper we propose two alternative techniques to cope with this problem:

a) Based on the minimum density in the feature (colour) space.
b) Based on maximum value of the density inside the current frame.

In the first proposed technique, the same colour density which is calculated recursively by equation (6) is used to identify a novel object. In this technique, out of the F_t pixels identified as a foreground in the current frame, t the one, $O_t^* = [h_t^*, v_t^*]$ which has minimum density (D), will be the most different from the background and most likely to represent the novel object/target on the image frame:

$$O_t^* = \arg\min_{t=1,i=1,j=1}^{N;H;V} D(z_t^{ij}) \tag{11}$$

It is a very fast technique and free of computational complexity. It is also guarantees a better lock on the object for tracking purposes (Fig. 3).

In the second alternative technique, we use again the density, but this time in terms of the spatial position of the pixels inside the current frame (for $i=1,2,\ldots,H;$ $j=1,2,\ldots,V$) which were identified already to be susceptive foreground, F_t. The pixel with maximum value of the density inside the current frame can be chosen to represent the novel object/target on the scene.

$$O_t^* = \arg\max_{i,j=1}^{F} \{D_t^{ij}\}, \quad O_t^* = [h_t^*, v_t^*] \tag{12}$$

where O_t^* denotes the vector of the object position in the current frame with its horizontal and vertical components. F denoted the number of pixels in a frame classified as foreground ($F << H*V$).

The rationale of this technique is that this point represents the pixel that has higher spatial density if only foreground (F_t) pixels are taken into account in the frame (Fig. 3).

The spatial density can be calculated recursively in a vector form similarly to (6)-(9):

$$D(O_t^*) = \frac{l-1}{(l-1)(f^T f + 1) - 2\gamma + \beta}; \; l=[1,F] \tag{13}$$

$$\gamma = f^T \delta \tag{14}$$

$$\beta(l) = \beta(l-1) + \|f(l-1)\|^2; \quad \beta(1) = 0 \tag{15}$$

$$\delta(l) = \delta(l-1) + f(l-1); \quad \delta(1) = 0 \tag{16}$$

where $f \in R^F$ denotes the vector of the foreground pixels in a frame

This method can be extended for image segmentation [14] , and landmark detection [9] used in self-localisation in robotics [10] . As result it is more robust to locate the position of the object in the current image frame compare to the standard mean value technique (Fig. 3).

3.2 A New Method for Multiple Objects Identification in Video Frames by Real-Time Clustering

Multiple objects tracking always has been a challenging part in computer vision. Several methods are used to identify and track the fixed number of objects [17]. Many of them are only applicable to tracking humans or vehicles [18-20]. The method that is proposed in this paper can be applied to tracking multiple objects whose number is unknown and varies during tracking. The proposal is for real-time on-line fast non-iterative clustering that does not require the number of clusters to be specified beforehand. This clustering approach is applied only to those pixels (F_t) in a frame that were identified as a novelty/foreground. In this approach the number of the clusters is not pre-specified and generated based on the position of the novelties in each frame. Each novelty/foreground is assigned

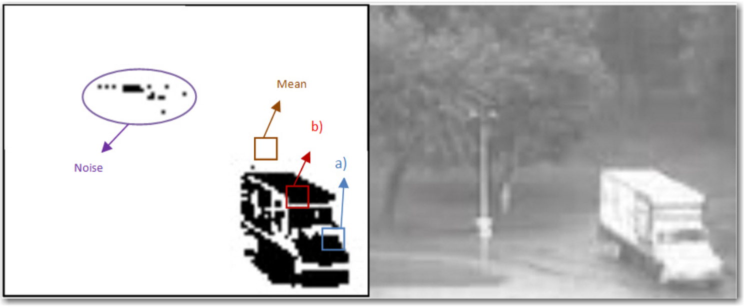

Fig. 3. Pixels detected as novelty; left hand side of the scene are due to noise and clutter; right hand side is modelled one. Note the pixels on the left hand side of the modelled scene are due to noise and clutter. The green and red square denotes the centre of the target as identified by the proposed techniques a) and b). Brown square denotes the centre as identified by the mean.

Fig. 4. Multi objects identification. Right hand side scenes are original frames; left hand side scenes are modelled ones. The red square denotes the focal point of the foreground.

to the cluster with the nearest mean. Initially a single cluster (object) is formed around the pixel identified as described in the previous section. Its radius, $r_1 = \sigma_1$ is determined based on the spatial variance of the positions of the pixels that are associated to it. After that, if the distance between the pixels that is a susceptive novelty/foreground and the centre of the cluster that is already formed is less than r_1, then a new cluster/target is created. At the same time in a pursuit to avoid noise and clutter we ignore the new clusters that are formed around a small number of pixels, $\underline{S}$. This number is determined in such a way that the size of an object/target that is expected to be detected to be comparable with the size of a regular (square) blob formed by $\underline{S}$ pixels. If any of the newly formed clusters has less than $\underline{S}$ pixels as members, it will be not specified as an object/target and will be ignored, see Fig. 4.

4 Real-Time Tracking Using Evolving Takagi-Sugeno eTS Fuzzy Systems

After detecting all the foreground pixels in an image frame and identifying the object/target often the problem is to track it. Therefore, the efficient tracking algorithm can be vital. In this paper we propose to use evolving Takagi-Sugeno (eTS) fuzzy model [12, 22] which represents a fuzzy mixture of locally active Kalman Filter (KF) [21] where the number of the local regions is not pre-specified and fixed. eTS is an on-line self-developing version of the widely used Takagi-Sugeno fuzzy systems [23] which combine a linguistic fuzzy *IF* part and a functional/linear *THEN* consequents part.

The proposed algorithm is non-linear with an evolving structure. This means that the number of local regions can grow or be reduced, the eTS structure – fuzzy rules, input variables/features, fuzzy sets – can expand or shrink according to the data pattern in the joint input-output (current – next frame) data space.

In a nutshell, learning eTS consists of two stages [24]:

1) Decomposing the data (pixel locations in the current and next/predicted images) space into local sub-areas.
2) Adapting the parameter of the consequent parts of the fuzzy rules.

Both stages are performed in real-time for an interval of time shorter than the time of arrival of the next image frame (less than *40ms* if assume *25fps* rate of the video).

In the tracking problem the aim is to predict the position of the object/target in the next, $(t+1)^{th}$ frame:

$$\hat{O}^{*}_{t+1} = eTS(O^{*}_{t}) \tag{17}$$

where $\hat{O}^{*}_{t+1} = eTS(O^{*}_{t})$ is the predicted position of the target in the $(t+1)^{th}$ frame.

Another advantage of eTS is that it can be represented by linguistically tractable fuzzy rules of the following type:

$$Rule: IF(h^{*}_{t} \text{ is about } \chi^{*}) \text{ AND } (v^{*}_{t} \text{ is about } w^{*}) \text{ THEN } \begin{cases} \hat{h}_{t+1} = a_0 + a_1 h_t + a_2 v_t \\ \hat{v}_{t+1} = b_0 + b_1 h_t + b_2 v_t \end{cases} \tag{18}$$

where χ^{*}, w^{*} are the prototypes (centres of the membership functions); a and b are the parameters of the (linear) consequents.

The fuzzy sets can defined by their membership functions, e.g. of a Gaussian type:

$$\mu^{l}_{\chi}(h) = e^{\frac{\left\| h^{l}_{t} - \chi^{*} \right\|}{2\sigma^{l}_{t}}} \tag{19}$$

where $\mu_{\chi}^{l}(h)$ denotes the membership to the fuzzy set $(h_t^* \text{ is about } \chi^*)$ form the l^{th} fuzzy rule.

Similar membership functions can be defined for the vertical component, v for each fuzzy rule, *l=[1,R]*. The overall prediction of the position of the target in the next frame is produced using centre of gravity type defuzzification:

$$\hat{O}^{*}_{t+1} = \sum_{l=1}^{R} \lambda^{l} O_t^{l*} \tag{20}$$

where $\lambda^l = \frac{\mu_{\chi}^{l}(h)\mu_{\omega}^{l}(v)}{\sum_{r=1}^{R}\mu_{\chi}^{r}(h)\mu_{\omega}^{r}(v)}$ is the normalized firing level of the l^{th} rule; $T_t^{l^*}$ is the prediction by the l^{th} rule.

Since eTS (same as TS) is linear in the consequents part it renders the use of well established learning approaches such as RLS. In fact, this is partially correct, because the linearity is correct only locally and eTS (same as TS) is non-linear as a whole. Therefore a fuzzily weighted version of RLS (wRLS) [24] is necessary to be applied, not the standard RLS. wRLS can be applied locally (per fuzzy rule and per cluster) or globally [13] . The local implementation has a number of advantages, including better convergence properties, better interpretability, smaller computational demands etc. [13] and is described as:

$$\hat{a}_t^{l} = \hat{a}_{t-1}^{l} + C_{t-1}^{l}\tau_{t-1}^{*}\lambda^{l}\left(\tau_t^{*} - \tau_{t-1}^{*T}\ \hat{a}_{t-1}^{l}\right);\ a_1^l = 0 \tag{21}$$

$$C_t^l = C_{t-1}^l - \frac{\lambda_i(\tau_{t-1})\ C_{t-1}^l\ \tau_{t-1}^{*}\ \tau_{t-1}^{*T}\ C_{t-1}^l}{1+\lambda_i(\tau_{t-1}^{*})\ \lambda_{t-1}^{*T}\ C_{t-1}^l\ \tau_{t-1}^{*}};\ C_1^l = \Omega I \tag{22}$$

where $\tau = [1; T^*]^T$ denotes the extended position vector; C is the co-variance matrix, λ is the normalized firing strength of the l^{th} fuzzy rule; Ω is a large value and I is an identity matrix.

5 Experimental Results

In this section, we present some real-time novelty detection results using the proposed RDE method. The RDE approach is applied to two video streams in different environment and illumination conditions (Fig. 5). The results are compared with the well known KDE approach in Table 1. Note that, the proposed algorithm is implemented in MATLAB, however the time can be significantly reduced using C language.

The first video sequence has 237 frames of size 176×144 while the second one is 320×240 with 195 frames. In both clips, the RDE method is applied for

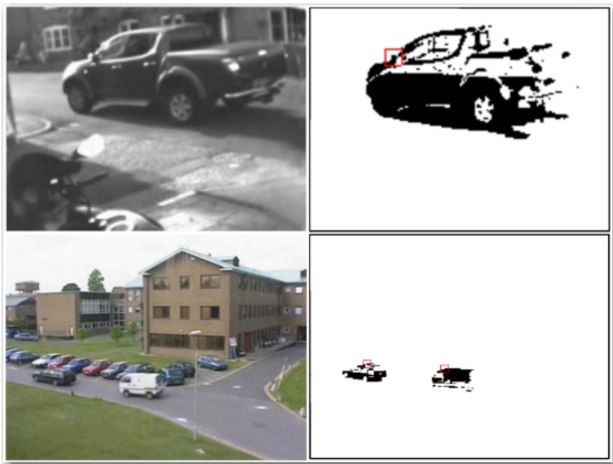

Fig. 5. Background Subtraction using the proposed RDE method, Left hand side scenes are original frames; right hand side scenes are modelled ones. The red square denotes the focal point of the foreground.

autonomous novelty detection. After novelty/foreground of the image frame is determined, two alternative novelty techniques known as minimum density in colour space and maximum spatial density (explained in section 3) are used to identify the single target, while the real-time clustering is implemented for automatic identification of multiple targets.

Despite the noise caused by illumination and camera oscillation in both video streams, the proposed approach (RDE) has a superior performance as compared to the original KDE approach. As opposed to KDE, the proposed algorithm is significantly faster and requires significantly less memory storage (Table 1). It should be stressed that RDE approach can be applied in real-time; while KDE approach is limited by the size of the window. If the size of the window is too large the sensitivity of the approach diminishes, on the contrary if the window size is too short it may lead to an oversensitive realization. Also, it is important to note that RDE can be realised on hardware (an n-line clustering approach using recursive Cauchy formula was implemented on FPGA [15] and proved to work extremely fast) which paves the way to various practical real-time implementations.

For tracking part, eTS was applied to predict the position of the target in the next t^{th} frame. At the end, the performance of the eTS was compared with KF in terms of two dimensions, h and v. Table 2 displays that eTS provides the smaller root mean error (RMSE) and non-dimensional index (NDEI) in estimating the true location of the target and overall has a better performance than KF.

In addition eTS provides the closer predicted values to the actual values comparing to KF (Fig. 6)

Table 1. Comparison of the performance of RDE and KDE in two different video streams

	Method	window size	Frame size (pixels)	Time per frame (sec)	Memory used Per pixel (bites)	Memory calculation
Video Clip (1)	KDE	20	176×144	15.57	60	O(60)
	RDE	1	176×144	0.43	4	O(4)
Video Clip (2)	KDE	20	320×240	44.56	60	O(60)
	RDE	1	320×240	1.17	4	O(4)

Table 2. Tracking precision using eTS vs. KF

	Method	RMSE	NDEI	VAF (%)
v, pixel	KF	15.58	0.7	50.57
	eTS	13.2	0.56	69.14
h, pixel	KF	23.78	0.66	55.58
	eTS	21.31	0.56	68.56

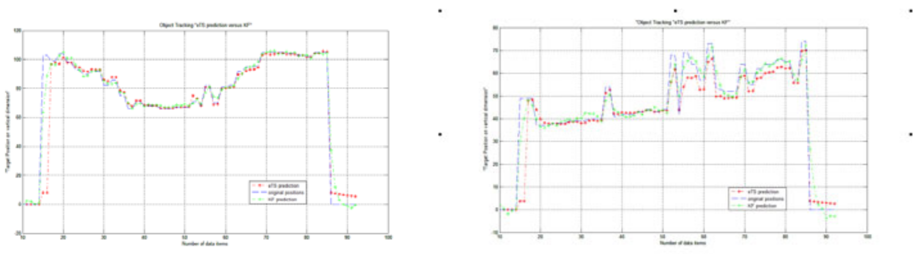

Fig. 6. Tracking performance of eTS vs. KF (Left plot – vertical component, Right plot – horizontal component)

6 Conclusion and Discussion

In this paper, we introduced novel techniques for detection and automatic object/target identification and tracking in video streams under uncertainties. We compared the results with the best known used methods such as kernel density estimation (KDE) for novelty detection and Kalman filter for tracking. The key innovation of the proposed approach is the use of a recursively calculated Cauchy type of kernel for the density estimation (as opposed to widely used Gaussian one) and in the tracking part of the problem – the use of evolving fuzzy Takagi-Sugeno model. The proposed approach is particularly suitable for real-time autonomous applications and is very fast and robust to uncertainties in the video stream (it does not need to be tuned to different environments and has built-in robustness to noise and clutter.

For single object/target identification, two alternative techniques based on minimum density in the colour across the video stream and the maximum spatial density inside the current frame was introduced and compared with are used and compared with spatial mean method. The results show a better lock on the target and more robust recognition comparing to the standard spatial mean technique. To autonomously identify multiple objects in a frame we proposed a real-time on-line clustering method which is computationally fast and the number of the clusters does not need to be pre-defined. For tracking autonomously identified objects/targets we proposed to use evolving Takagi-Sugeno fuzzy model (eTS) which provides real-time high prediction, is fast and human-interpretable. The overall proposed approach is fully autonomous and suitable for video-analytical tasks in surveillance and autonomous systems design.

References

1. Hampapur, A.: Smart Video Surveillance, Exploring the concept of multi-scale spatiotemporal tracking. IEEE Signal Processing Magazine, 38–51 (2005)
2. Han, B., Comaniciu, D., Davis, L.: Sequential kernel density approximation through mode propagation: applications to background modeling. In: Proc. ACCV - Asian Conf. on Computer Vision (2004)
3. Gonzalez, R.C., Woods, R.E.: Digital Image Processing, 2nd edn. Prentice Hall, Englewood Cliffs (2002) ISBN: 0201180758
4. Elgammal, A., Duraiswami, R., Harwood, D., Davis, L.: Background and Foreground modeling using nonparametric Kernel Density Estimation for visual surveillance KDE. Proc. 2002 IEEE 90(7), 1151–1163 (2002)
5. Cheung, S.-C., Kamath, C.: Robust techniques for background subtraction in urban traffic video. In: Proc. SPIE, Electronic Imaging Video Comm. and Image Proc., San Jose, pp. 881–892 (2004)
6. Cucchiara, R., Grana, C., Piccardi, M., Prati, A.: Detecting moving objects, ghosts and shadows in video streams. IEEE Trans. on Pattern Analysis and Machine Intelligence, 1337–1342 (2003)
7. Han, B., Comaniciu, D., Davis, L.: Sequential kernel density approximation through mode propagation: applications to background modeling. In: Proc. ACCV - Asian Conf. on Computer Vision (2004)
8. Memon, A., Angelov, P., Ahmed, H.: An Approach to Real-Time Color-based Object Tracking. In: Proc. 2006 Intern. Symp. on Evolving Fuzzy Systems, Ambleside, Lake District, UK, pp. 81–87. IEEE Press, Los Alamitos (2006)
9. Zhou, X., Angelov, P.: Real-Time joint Landmark Recognition and Classifier Generation by an Evolving Fuzzy System. In: Proc. 2006 World Congress on Computational Intelligence, Vancouver, Canada, pp. 6314–6321 (2006)
10. Zhou, X., Angelov, P.: Autonomous Visual Self-localization in Completely Unknown Environment using Evolving Fuzzy Rule-based Classifier. In: Proc. IEEE Intern. Conf. on Comp. Intel. Applic. for Defense and Security, Honolulu, USA, pp. 131–138 (2007)
11. Zhivkovic, Z., Van der Heijden, F.: Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recognition Letters, 773–780 (2006)
12. Angelov, P.: An Approach for Fuzzy Rule-base Adaptation using On-line Clustering. Intern. Journal of Approximate Reasoning, 275–289 (2004)

13. Angelov, P., Filev, D.: An Approach to On-line Identification of Takagi-Sugeno Fuzzy Models. IEEE Trans. on System, Man, and Cybernetics, part B - Cybernetics, 484–498 (2004)
14. Alzate, C., Suykens, J.: Image segmentation using Weighted Kernel PCA Approach to Spectral Clustering. In: Proc. IEEE Intern. Conf. on Comp. Intell. Applicat. for Image and Signal Processing, CIISP 2007, Honolulu, HI, USA, pp. 220–225 (2007)
15. Everett, M., Angelov, P.: EvoMap: On-Chip Implementation of Intelligent Information Modelling using EVOlving MAPping. Technical Report, Lancaster University, UK, pp. 1–15 (2005)
16. Cucchiara, R., Grana, C., Piccardi, M., Prati, A.: Detecting moving objects, ghosts and shadows in video streams. IEEE Trans. on Patt. Anal. and Machine Intell. 25, 1337–1342 (2003)
17. MacCormick, P.J., Blake, A.: A probabilistic exclusion principle for tracking multiple objects. In: ICCV 1999, pp. 572–578 (1999)
18. Mao, X., Qi, F., Zhu, W.: Multiple-part based Pedestrian Detection using Interfering Object Detection. In: Third International Conference on Natural Computation, pp. 165–169 (2007)
19. Cai, Q., Aggarwal, J.K.: Tracking human motion using multiple cameras. In: 13th International Conference on Pattern Recognition, pp. 68–72 (1996)
20. Yang, L., Johnstone, J., Zhang, C.: A Multi-camera Approach to Vehicle Tracking Based on Features. In: IEEE International Symposium on Multimedia, pp. 79–80 (2007)
21. Kalman, R.E.: A New Approach to linear filtering and prediction problem. Trans. of the ASME, Ser. D, Journal of Basic Engineering, 34–45 (1960)
22. Angelov, P., Zhou, X.: Evolving fuzzy systems from data streams in Real time. In: Proc. 2006 Intern. Symp. on Evolving Fuzzy Systems, Ambleside, Lake District, UK, pp. 29–35. IEEE Press, Los Alamitos (2006)
23. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its application to modelling and control. IEEE Trans. on Systems, Man and Cybernetics, 116–132 (1985)
24. Angelov, P.: Evolving Rule-based Models: A Tool for Design of Flexible Adaptive Systems. Springer, Heidelberg (2002)