Spatio-Temporal Nonparametric Background Modeling and Subtraction

Raviteja Vemulapalli R. Aravind
Department of Electrical Engineering
Indian Institute of Technology, Madras, India.

Abstract

Background modeling and subtraction is a core component of many vision based systems. By far the most popular background models are per-pixel models, in which each pixel is considered independently. Such models fail to handle dynamic backgrounds and noise. In this paper, we present a solution to this problem by proposing a novel and computationally simple spatio-temporal background model. We extend the nonparametric background model [5], one of the most widely used per-pixel models, from temporal domain to spatio-temporal domain. Instead of individual pixels, we consider 3×3 blocks centered on each pixel and use kernel density estimation (KDE) method in the 9-dimensional space. In order to reduce the computational complexity we use a hyperspherical kernel instead of Gaussian. We also make a small modification to the short term model used in [5] in order to handle sudden illumination changes. Experimental results show the effectiveness of the proposed model.

1. Introduction

The separation of moving objects (foreground) from rest of the scene (background) in a video is a very important task in many computer vision based applications such as object identification and tracking, human activity recognition, crowd density and behaviour estimation, traffic monitoring, etc. The most widely used technique for this purpose in case of static camera applications is background subtraction, in which a model representing the background is built and regularly updated with the changes in the background. Once the model is built, each pixel in the current frame is classified as background or foreground using the current background model. The most important and difficult part of background subtraction is building a background model. Various background models have been proposed in the past. They can be broadly classified into *per-pixel* or *temporal models* and *block-based* or *spatio-temporal models*.

1.1. Per-pixel or Temporal models

Per-pixel models maintain a separate statistical model for each pixel and the model for a pixel is learned entirely from the history of that pixel alone. In [1] Wren *et al*. modeled each pixel using a single Gaussian whose mean and variance were regularly updated. This model works for static or slowly changing backgrounds but fails in case of dynamic backgrounds. Stauffer and Grimson used a mixture of fixed number of adaptive Gaussians in [2] to capture multimodal nature of the background and in [3] Shimada *et al*. proposed a fast method in which the number of Gaussians for each pixel can be changed dynamically to adapt to the changes in the background. Porikli and Tuzel modeled each pixel as a set of layered normal distributions in [4] and used Bayesian learning method not only to estimate the mean and variance of each layer but also the probability distributions of the mean and variance.

In [5] Elgammal *et al*. proposed a model in which Gaussian kernel was used to estimate the intensity PDF of a pixel from its past samples. Then the pixel was classified as background or foreground by putting a threshold on the value of estimated PDF at current intensity value. Later Tanaka *et al*. used a rectangular kernel instead of Gaussian in [6] which decreased the computational cost and reduced the probability density estimation into an incremental process. Mittal and Paragios used motion information for background-foreground differentiation in [7]. Though this method is able to handle dynamic backgrounds like waving trees and ocean waves, it is computationally highly complex.

Ridder *et al*. used Kalman filtering for background modeling in [8] whereas Toyama *et al*. used Wiener filtering in [9] to linearly predict the intensity value of a pixel. A pixel was classified as background or foreground based on the deviation of its actual intensity value from the predicted value. In [10] Haritaoglu *et al*. modeled the background for each pixel using the maximum and minimum intensity values of that pixel and the maximum inter-frame change in intensity at that pixel observed during training phase. This model fails when the background pixels are multimodal distributed or widely dispersed in intensity.

2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops
978-1-4244-4441-0/09/$25.00 ©2009 IEEE

1.2. Block-based or Spatio-temporal models

In case of block-based models, the background model of a pixel depends not only on that pixel but also on the near-by pixels. In [11] Oliver *et al*. considered the whole image as a single block and used the best M eigenvectors generated by applying PCA to a set of training images to represent the background. In [12] Monnet *et al*. divided each frame into blocks and then mapped each block into a lower dimensional feature space whose basis vectors were incrementally updated. A prediction mechanism was used in the lower dimensional feature space for background-foreground differentiation. In [13] Seki *et al*. proposed a background subtraction method in which the frames were divided into blocks and co-occurrences of image variations at neighbouring blocks were used for dynamically narrowing the permissible range of background image variations. One major disadvantage of these block-based methods is that the boundary of the foreground objects cannot be delineated exactly.

In recent years researchers have been concentrating more on incorporating spatial aspect into background modeling to take advantage of the correlation that exists between neighbouring pixels. In [14] Pless used a mixture-of-Gaussians distribution for each pixel in the feature space defined by intensity and the spatio-temporal derivatives of intensity at that pixel. In [15] Babacan and Pappas used a spatio-temporal hybrid model. Gibbs-Markov random field was used to model spatial interactions and Gaussian mixture model was used to model temporal interactions.

Last but not least, some researchers have also used texture based methods to incorporate spatial aspect into background modeling. In [16] Heikkila *et al*. modeled each pixel as a group of adaptive local binary pattern (LBP) histograms and in [17] Zhang *et al*. extended the local binary patterns from spatial to spatio-temporal domain and then modeled each pixel as a group of STLBP (spatio-temporal local binary pattern) histograms.

In this paper, we first extend the nonparametric background model proposed in [5] from temporal domain to spatio-temporal domain, to propose a new spatio-temporal background model, which we call as spatio-temporal nonparametric background model. Then to reduce the computational complexity we propose to use a hyperspherical kernel instead of Gaussian for probability density estimation. Compared to the temporal nonparametric model [5], the proposed model is able to handle dynamic backgrounds and noise very well.

The remainder of this paper is organized as follows. Section 2 discusses the temporal nonparametric model [5]. In section 3 we extend it to propose a new spatio-temporal background model. We present our experimental results in section 4 and conclude the paper in section 5.

2. Temporal nonparametric model

2.1. Basic model

Let $I(x,y)$ denote the intensity value of pixel (x,y) in the current frame and the set of its past intensity samples be $\{I_1(x,y), I_2(x,y), ..., I_N(x,y)\}$. Then the current intensity PDF of the pixel (x,y) can be nonparametrically estimated from its past intensity samples using a kernel estimator function K as

$$p\left(z \mid \{I_i(x,y)\}_{i=1}^{N}\right) = \frac{1}{N}\sum_{i=1}^{N} K(z - I_i(x,y)) \qquad (1)$$

Let the value of this intensity PDF at $z = I(x,y)$ be p'.

$$p' = p\left(I(x,y) \mid \{I_i(x,y)\}_{i=1}^{N}\right) \qquad (2)$$

Using a Gaussian $N(0,\Sigma)$ as the kernel estimator function K, where Σ represents the kernel function bandwidth, the probability density p' becomes

$$p' = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}[I(x,y) - I_i(x,y)]^T \, \Sigma^{-1} \, [I(x,y) - I_i(x,y)]\right\} \qquad (3)$$

where d denotes the number of dimensions (1 for gray-scale and 3 for colour videos).

If different colour channels are assumed to be independent then

$$\Sigma = \begin{pmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \sigma_2^2 & 0 \\ 0 & 0 & \sigma_3^2 \end{pmatrix} \qquad (4)$$

and the probability density becomes

$$p' = \frac{1}{N}\sum_{i=1}^{N}\prod_{j=1}^{3} \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left\{-\frac{[I^j(x,y) - I_i^j(x,y)]^2}{2\sigma_j^2}\right\} \qquad (5)$$

where $I_i^j(x,y)$ denotes the j^{th} colour component of $I_i(x,y)$.

Once the probability density is estimated, the pixel (x,y) is classified as foreground or background based on whether the estimated density is below or above a threshold value respectively.

2.2. Updating the background

Given a new pixel sample, there are two mechanisms to update the sample set.

- **Selective update**: New sample is added to the sample set only if it belongs to background.
- **Blind update**: New sample is added to the sample set irrespective of whether it belongs to background or foreground.

In both the cases when a new sample is added, the oldest sample is removed from the sample set to ensure that the probability density estimation is based on recent samples. There are tradeoffs to both these mechanisms and to avoid them two models (short term model and long term model) were proposed and a combination of both was used in [5].

- **Short term model**: This is a very recent model of the scene. It uses the most recent N background sample values. The sample set is updated using selective update mechanism.
- **Long term model**: This model captures a more stable representation of the background. It consists of N samples taken from a much larger window W in time. The sample set is updated using blind update mechanism.

2.3. Foreground detection

Results of both short term and long term models were used in classifying a pixel as foreground or background.

Table 1: Combination results
(0 for background and 1 for foreground)

Short term model	Long term model	Final result
$O_s(x,y)=0$	$O_l(x,y)=0$	$O(x,y)=0$
$O_s(x,y)=0$	$O_l(x,y)=1$	$O(x,y)=0$
$O_s(x,y)=1$	$O_l(x,y)=0$	$O(x,y)=O'(x,y)$
$O_s(x,y)=1$	$O_l(x,y)=1$	$O(x,y)=1$

Table 1 summarizes the combination of short term and long term models used for foreground detection in [5]. $O_s(x,y)$, $O_l(x,y)$ and $O(x,y)$ denote the short term model detection result, the long term model detection result and the final result respectively for pixel (x,y) and $O'(x,y)$ is given by

$$O'(x,y) = \begin{cases} 1 & \text{if } \sum_{i=-1}^{1}\sum_{i=-1}^{1} O_s(x-i,y-j)O_l(x-i,y-j) \neq 0 \\ 0 & \text{else} \end{cases} \quad (6)$$

3. Proposed model

In the previous section we have discussed the temporal nonparametric background model, which is one of the most widely used per-pixel background models. In this section we extend it to spatio-temporal domain. In the case of per-pixel background models erroneous detection rate can be potentially high due to intensity value of a foreground pixel falling by chance within the range associated with background or intensity of a background pixel sporadically assuming a value which is not associated with background. This happens more frequently in case of gray-scale videos and noisy videos. One way to address this issue is to use the information from neighbouring pixels while classifying a pixel as foreground or background. So in our model instead of considering each pixel independently we consider 3×3 blocks centered on each pixel and develop a block-based background model to take advantage of the correlation that exists between neighbouring pixels.

3.1. Basic model for gray-scale videos

Similar to the temporal nonparametric model we use N past frames to segment the current frame into background and foreground. Let I denote the current frame and $I_1, I_2, ..., I_N$ denote the N past frames being used to segment the current frame. Consider a pixel (x,y). We denote the 3×3 block centered on (x,y) in I by $F(x,y)$. Similarly the 3×3 blocks centered on (x,y) in $I_1, I_2, ..., I_N$ are denoted by $F_1(x,y), F_2(x,y), ..., F_N(x,y)$. Each $F_i(x,y)$ is a 9-dimensional data point. In our model instead of considering $I(x,y)$ and $\{I_1(x,y), I_2(x,y), ..., I_N(x,y)\}$, we consider $F(x,y)$ and $\{F_1(x,y), F_2(x,y), ..., F_N(x,y)\}$ and use the kernel density estimation technique in the 9-dimensional space. The current intensity PDF of the 3×3 block centered on (x,y) can be nonparametrically estimated from its past intensity samples using a kernel estimator function K as

$$p\left(Z \mid \{F_i(x,y)\}_{i=1}^{N}\right) = \frac{1}{N}\sum_{i=1}^{N} K(Z - F_i(x,y)) \quad (7)$$

Let the value of PDF at $Z = F(x,y)$ be f.

$$f = p\left(F(x,y) \mid \{F_i(x,y)\}_{i=1}^{N}\right) \quad (8)$$

If we use a Gaussian kernel like in [5] then the probability density is given by

$$f = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}[F(x,y) - F_i(x,y)]^T \Sigma^{-1} [F(x,y) - F_i(x,y)]\right\} \quad (9)$$

where d is the number of dimensions, which is 9 in this case. Presence of 9-dimensional Gaussians makes this model computationally highly complex and difficult for real time implementation.

In order to reduce the computational cost, we propose to use a hyperspherical kernel in the 9-dimensional space. Hyperspherical kernel in a d-dimensional space can be represented as

$$K_c(\mathbf{u}) = \begin{cases} \frac{1}{V} & \text{if } |\mathbf{u}| \leq r \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where r is the radius or bandwidth of the kernel and V is the volume of a hypersphere with radius r in the d-dimensional space.

Now using a hyperspherical kernel of radius r the probability density f can be estimated as

$$\begin{aligned} f &= \frac{1}{N}\sum_{i=1}^{N} K_c(F(x,y) - F_i(x,y)) \\ &= \frac{M}{N\,V} \end{aligned} \tag{11}$$

where M is the number of past intensity samples lying inside the hypersphere of radius r centered on $F(x,y)$ in the 9-dimensional space. M can be calculated using

$$M = \sum_{i=1}^{N} \phi\left(\frac{\|F(x,y) - F_i(x,y)\|}{r}\right) \tag{12}$$

where

$$\phi(u) = \begin{cases} 1 & \text{if } u \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

and $\|F(x,y) - F_i(x,y)\|$ denotes the euclidean distance between the data points $F(x,y)$ and $F_i(x,y)$.

Once the probability density is estimated, we compare it with a threshold *t*. If the estimated density f is above/below the threshold value then the centre pixel (x,y) is classified as background/foreground.

One more issue to be addressed here is the calculation of volume V of the hypersphere, which is a function of the kernel radius r.

$$\frac{M}{N\,V} \leq t \iff \frac{M}{N} \leq thr \tag{14}$$

where $thr = tV$. So we need not explicitly calculate V. It is enough if we adjust *thr* based on the kernel radius r.

3.2. Foreground detection

Similar to the temporal nonparametric model, we use both long term model and short term model (with a small modification) and follow the same rules for detecting the foreground and updating the background.

In the short term model used in [5] selective update mechanism was used. In case of sudden illumination change most of the frame will be declared as foreground and will remain as foreground till the long term model adapts itself to the new lighting conditions. To avoid this, when more than certain percentage α of the frame is declared as foreground, we consider it as sudden illumination change and use blind update mechanism for the short term model. If blind update mechanism is used for the short term model, it adapts to the new lighting conditions quickly and avoids the detection of background as foreground.

3.3. Model for colour videos

In case of colour videos, we assume independence between different colour channels and hence process each channel independently like a gray-scale video. Let $R(x,y),\ G(x,y),\ B(x,y)$ be the binary (0 for background and 1 for foreground) variables indicating the processing results of the three colour channels at pixel (x,y). Then the final result $O(x,y)$ is given by

$$O(x,y) = R(x,y) \parallel G(x,y) \parallel B(x,y) \tag{15}$$

where $\parallel$ denotes logical OR.

By using a hyperspherical kernel instead of Gaussian kernel the probability density estimation reduces to just finding the number of past samples lying inside the hypersphere of radius r centered on $F(x,y)$, thus making it computationally simple and easy-to-implement. But there is a tradeoff between computational efficiency and foreground detection. Hyperspherical kernel is a discontinuous kernel whereas Gaussian kernel is a continuous one. So Gaussian kernel gives a better estimate of the probability density and hence better foreground detection results compared to hyperspherical kernel. But our experiments show that even with hyperspherical kernel we can get very good results.

The kernel radius or bandwidth r is a very important parameter that influences the estimated probability density and hence the detection results. If r is too large, then the estimated PDF becomes too smooth and the probability that the foreground pixels are missed becomes high. On the other hand, if r is too small, the estimated function is not smooth, which means that the estimated density value is strongly dependent on the observed pixel values and the probability that the background pixels are misclassified as foreground pixels becomes very high. Value of the parameter r has to adjusted depending on the amount of variation present in the background.

4. Experimental results

To show the effectiveness of the proposed spatio-temporal nonparametric model we conducted experiments using four different image sequences. The first two are noisy sequences and the next two correspond to dynamic background. We used a combination of long term and short

term models for both temporal nonparametric and spatio-temporal nonparametric background models in our experiments. For short term model we used the most recent 50 background samples ($N = 50$) and for long term model we used 50 samples selected from the most recent 250 samples irrespective of whether they belong to foreground or background ($W = 250$). The value of parameter α is taken as 75%.

The first sequence we considered is lab sequence, in which two persons walk in opposite directions inside a lab. This sequence consists of 445 gray-scale images of size 280×350. To show the effectiveness of the proposed model in the presence of noise, we added three levels of Gaussian noise corresponding to the standard deviations of 10, 25 and 50. Figure 1 shows the detection results for 346^{th} and 375^{th} frames of this sequence under different noise levels. In the case of temporal nonparametric model many background pixels are misclassified as foreground when the threshold is high. As the threshold is reduced to suppress the background, many foreground pixels are misclassified as background (observe that some of the foreground objects almost disappear in the third column of the results). The proposed model is able to avoid such misclassifications to a great extent and segment out the foreground objects even when the video is highly corrupted by noise.

The second sequence we considered is crowd sequence, in which 16 people walk randomly on an outdoor lawn. This sequence consists of 400 RGB images of size 200×320. We added Gaussian noise with a standard deviation of 25 to this image sequence. Figure 2 shows the detection results for 332^{nd}, 350^{th} and 381^{st} frames of this sequence. Our results once again show that the proposed model gives much better results compared to the temporal nonparametric model in case of noisy sequences.

The third sequence we considered is a video of a floating bottle. This sequence consists of 340 gray-scale images of size 120×160. This is a challenging sequence as the background (waves on the surface of water) in this case is dynamic. Figure 3 shows the detection results for 291^{st}, 302^{nd} and 328^{th} frames of this sequence. We can clearly see that the proposed model is able to classify the waves as background without much loss in the foreground (bottle) whereas the temporal nonparametric model fails to do so. When threshold is high it classifies waves as foreground and when threshold is reduced to suppress the waves some part of foreground is lost.

The fourth sequence we considered is a video of ducks swimming in a river. This sequence consists of 375 RGB images of size 200×320. This sequence also corresponds to dynamic background (swinging grass and running water). Figure 4 shows the detection results for 292^{nd}, 340^{th} and 363^{rd} frames of this sequence. For this sequence also the proposed model is able to segment out the foreground objects (moving ducks) from the dynamic background whereas the temporal nonparametric model fails to do so.

When implemented in MATLAB the processing speed of the proposed model was about one-third of the processing speed of the temporal nonparametric model. Memory requirement of the proposed model is 5 times the memory needed for the temporal nonparametric background model. The long term model needs same amount of memory in both the cases whereas the short term model memory requirements for the proposed model and the temporal nonparametric model are in the ratio 9:1.

Our experimental results show that the proposed spatio-temporal nonparametric model gives superior performance compared to the temporal nonparametric model in case of noisy videos and dynamic backgrounds. Many block based models [11, 12, 13] have been proposed earlier to incorporate spatial aspect into background modeling. One major disadvantage of such models is that the detected foreground is a combination of rectangular boxes and hence information on shape of the foreground objects is lost. Though the proposed model is also a block-based model, it doesn't suffer from this loss as the result obtained by processing a block is applied only to the center pixel. We can clearly see from the results that the shape information of foreground objects is preserved in our approach.

5. Conclusion and future work

In this paper we extended the temporal nonparametric background model [5] to propose a novel and computationally simple spatio-temporal background model, which we call as spatio-temporal nonparametric background model. Instead of considering each pixel independently, we considered 3×3 blocks centered on each pixel and used KDE method in the 9-dimensional space. In order to reduce the computational cost associated with Gaussian kernel, we used hyperspherical kernel in the 9-dimensional space. We also made a small modification to the short term model proposed in [5] in order to handle sudden illumination changes. Our experimental results show that the proposed model is able to handle noise and dynamic backgrounds very well. Another important advantage of our approach is that it retains information regarding the shape of foreground objects unlike many earlier proposed block based models.

In our model we assumed different colour channels to be independent. For future work, one can consider all the three colour channels together resulting in 27-dimensional data. To avoid dealing with 27-dimensional data one can use incremental PCA to reduce the dimensionality and update the basis vectors incrementally. In our approach we used fixed radius for the hyperspherical kernel and fixed threshold value. One can work in the direction of making these parameters adaptive.

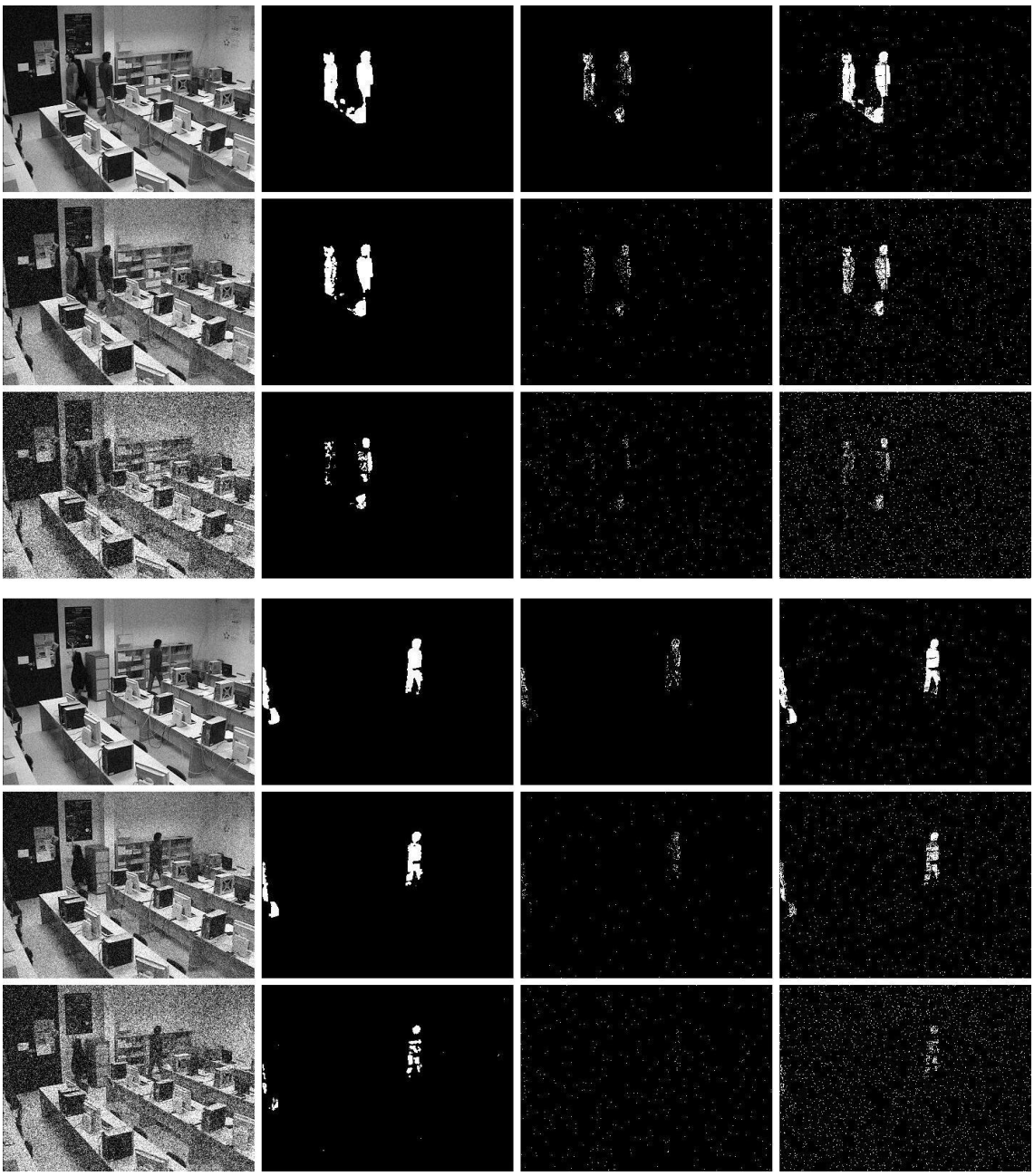

Figure 1. Detection results for 346^{th} (first three rows) and 375^{th} (remaining three rows) frames of lab sequence under different noise levels. First and fourth rows correspond to low level noise ($\sigma = 10$), second and fifth rows correspond to medium level noise ($\sigma = 25$), third and sixth rows correspond to high level noise ($\sigma = 50$). The first column shows the original frames corrupted by noise, the second column shows the results of the proposed model, the third and fourth columns show the results of the temporal nonparametric model for low and high threshold values respectively.

References

[1] C. Wren, A. Azarbayejani, T. Darrell and A. Pentland, "Pfinder: Real-Time Tracking of the Human Body", in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 780-785, July 1997.

[2] C. Stauffer and W. E. L. Grimson, "Adaptive Background Mixture Models for Real-time Tracking", in *IEEE Conference on Computer Vision and Pattern Recognition*, pages 246-252, 1999.

[3] A. Shimada, D. Arita and R. Taniguchi, "Dynamic control of adaptive mixture-of-Gaussians background model", in *IEEE International Conference on Advanced Video and Signal based Surveillance*, Nov 2006.

Figure 2. Detection results for 332^{nd}, 350^{th} and 381^{st} frames of crowd sequence from top row to bottom row respectively. The first column shows the original frames corrupted by noise ($\sigma = 25$), the second column shows the results of the proposed model, the third and fourth columns show the results of the temporal nonparametric model for low and high threshold values respectively.

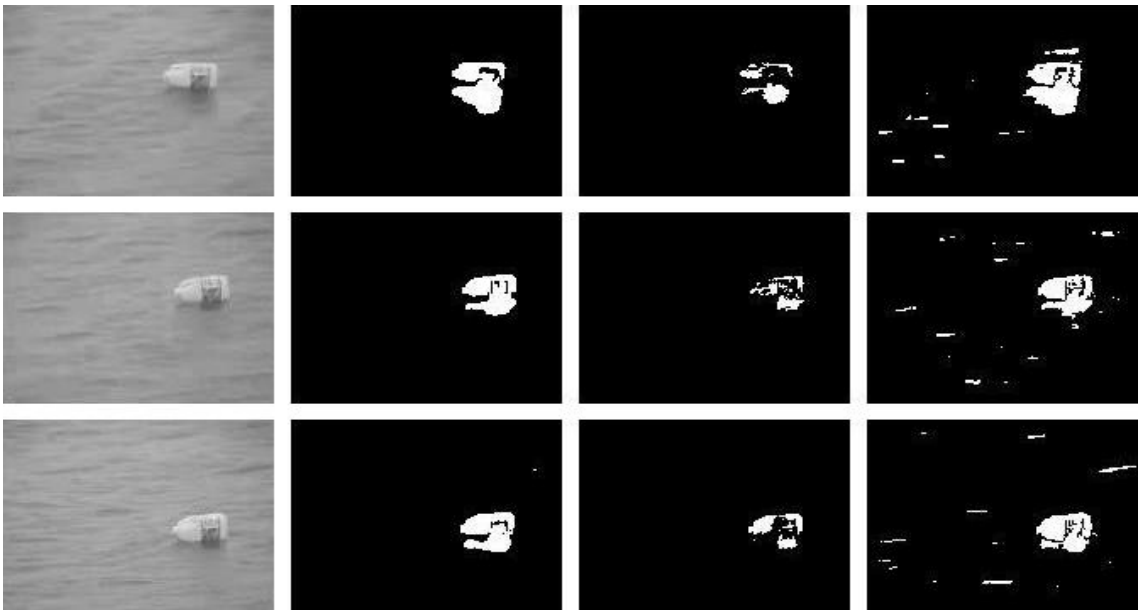

Figure 3. Detection results for 291^{st}, 302^{nd} and 328^{th} frames of bottle sequence from top row to bottom row respectively. The first column shows the original frames, the second column shows the results of the proposed model, the third and fourth columns show the results of the temporal nonparametric model for low and high threshold values respectively.

[4] F. Porikli and O. Tuzel, "Bayesian background modeling for foreground detection", in *International Workshop on Visual Surveillance and Sensory Networks*, pages 55-58, 2005.

[5] A. Elgammal, D. Harwood and L. Davis, "Nonparametric model for background subtraction", in *European conference on Computer Vision*, pages 751-767, Dublin, Ireland, May 2000.

[6] T. Tanka, A. Shimada, D. Arita, R. Taniguchi, "A fast algorithm for adaptive background model construction using parzen density estimation", in *IEEE International Conference on Advanced Video and Signal based Surveillance*, pages 528-533, London, Sep 2007.

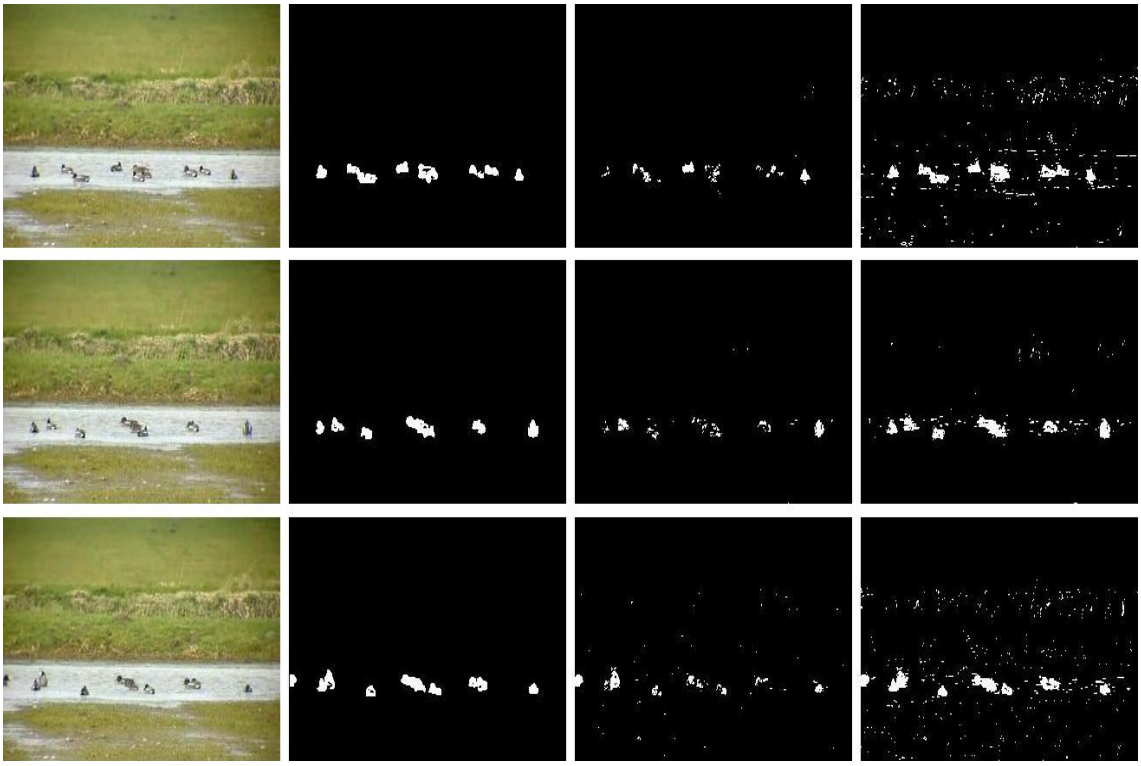

Figure 4. Detection results for 292^{nd}, 340^{th} and 363^{rd} frames of ducks sequence from top row to bottom row respectively. The first column shows the original frames, the second column shows the results of the proposed model, the third and fourth columns show the results of the temporal nonparametric model for low and high threshold values respectively.

[7] A. Mittal and N. Paragios, "Motion based background subtraction using adaptive kernel density estimation", in *IEEE Conference on Computer Vision and Pattern Recognition*, volume II, pages 302-309, Washington, DC, 2004.

[8] C. Ridder, O. Munkelt and H. Kirchner, "Adaptive background estimation and foreground detection using kalman filtering", in *International Conference on Recent Advances in Mechatronics*, pages 193-199, 1995.

[9] K. Toyama, J. Krumm, B. Brumitt and B. Meyers, "Wallflower: Principles and practice of background maintenance", in *International Conference on Computer Vision*, pages 255-261, Kerkyra, Greece, September 1999.

[10] Haritaoglu, I. D. Harwood, L. S. Davis, "W^4 Who? When? Where? What? A real time system for detecting and tracking people", in 5^{th} *European Conference on Computer Vision*, 1998, Freiburg, Germany.

[11] N. M. Oliver, B. Rosario and A. P. Pentland, "A Bayesian computer vision system for modeling human interactions", in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):831-843, Aug 2000.

[12] A. Monnet, A. Mittal, N. Paragios and V. Ramesh, "Background modeling and subtraction of dynamic scenes", in *International Conference on Computer Vision*, pages 1305-1312, Nice, France, Oct 2003.

[13] M. Seki, T. Wada, H. Fujiwara and K. Sumi, "Background subtraction based on co-occurrence of image variation", in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 65-72, Madison, Wisconsin, June 2003.

[14] R. Pless, "Spatio-temporal background models for outdoor surveillance", in *Journal on Applied Signal Processing*, pages 2281-2291, 2005.

[15] S. D. Babacan and T. N. Pappas, "Spatio-temporal algorithm for joint video segmentation and foreground detection", in *Proceedings of European Signal Processing Conference*, Florence, Italy, Sep 2006.

[16] M. Heikkila, M. Pietikainen and J. Heikkila, "A texture based method for modeling the background and detecting moving objects", in *Proceedings of British Machine Vision Conference*, volume 1, pages 187-196, 2004.

[17] S. Zhang, H. Yao and S. Liu, "Dynamic background modeling and subtraction using spatio-temporal local binary patterns", in *IEEE International Conference on Image Processing*, San Deigo, California, 2008.