Intelligent pre-processing for fast-moving object detection

Chris Poppe, Sarah De Bruyne, Gaëtan Martens, Peter Lambert, and Rik Van de Walle

Department of Electronics and Information Systems - Multimedia Lab, Ghent University - IBBT, Gaston Crommenlaan 8 Bus 201, B-9050 Ledeberg-Ghent, Belgium

ABSTRACT

Detection and segmentation of objects of interest in image sequences is the first major processing step in visual surveillance applications. The outcome is used for further processing, such as object tracking, interpretation, and classification of objects and their trajectories. To speed up the algorithms for moving object detection, many applications use techniques such as frame rate reduction. However, temporal consistency is an important feature in the analysis of surveillance video, especially for tracking objects. Another technique is the downscaling of the images before analysis, after which the images are up-sampled to regain the original size. This method, however, increases the effect of false detections. We propose a different pre-processing step in which we use a checkerboard-like mask to decide which pixels to process. For each frame the mask is inverted to avoid that certain pixel positions are never analyzed. In a post-processing step we use spatial interpolation to predict the detection results for the pixels which were not analyzed. To evaluate our system we have combined it with a background subtraction technique based on a mixture of Gaussian models. Results show that the models do not get corrupted by using our mask and we can reduce the processing time with over 45% while achieving similar detection results as the conventional technique.

Keywords: Moving object detection, Mixture of Gaussian Models, video surveillance

1. INTRODUCTION

Video surveillance is proliferating worldwide, and although the efforts done to create smart autonomic video surveillance systems are increasing, providing fast and accurate solutions remains difficult. The recent rapid increase in the amount of surveillance cameras has led to a strong demand for automatic methods for processing their outputs. Typical surveillance systems start with the detection and extraction of moving objects in image sequences. Next, these objects are tracked during time and are classified in known categories. Subsequently, intelligent decisions about the behaviour of these objects can be taken and alerts are issued if necessary. Since automation is the goal, it is desirable to achieve very high accuracy with the lowest possible false alarm rates.

The detection of moving objects in complex environments has been the research subject in the computer vision community for several years now and different approaches are developed.[1] One of these approaches is background subtraction. During the surveillance of a scene, a reference background model is built and dynamically updated to represent the environment. New images are compared with this model to yield regions of differences, called foreground regions. Many different models have been proposed for background subtraction, of which the Mixture of Gaussian Models (MGM) is one of the most popular.[2] Stauffer and Grimson proposed to model the value of each pixel as a mixture of Gaussians and use an approximation technique to update the model. Recently, the MGM method is showing tremendous popularity thanks to the dynamic and multimodal behaviour. One drawback that MGM suffers from, is the high computational cost to maintain the different Gaussian models.

In this paper, we present a pre-processing step that can increase the speed of typical background subtraction systems without significantly reducing the detection accuracy. We propose the use of an analysis mask to decide which pixels should be fully analyzed. The detection result by the background subtraction technique of these

C.P.: chris.poppe@ugent.be, phone: +32 (0)9 33 14959
S.DB.: sarah.debruyne@ugent.be, phone: +32 (0)9 33 14957
G.M.: gaetan.martens@ugent.be, phone: +32 (0)9 33 14959
P.L.: peter.lambert@ugent.be, phone: +32 (0)9 33 14 993
R.VdW: rik.vandewalle@ugent.be, phone: +32 (0)9 33 14 911

pixels are then interpolated to obtain an estimation for the surrounding pixels. We apply it to MGM and present our results.

The next section elaborates on related work. Next, the Mixture of Gaussian Models is discussed more thoroughly. Subsequently, our proposed changes to the original scheme are presented and experimental results are provided. Finally, we end with the conclusions.

2. RELATED WORK

To provide sufficiently detailed detection, typical background subtraction systems will maintain one or more background models for each pixel in the image. Consequently, newly captured images need to be analyzed entirely. This means that every new pixel has to be checked for consistency with the specific background model. Afterwards, a decision is taken for each pixel whether it represents background or foreground. Finally, adaptive background subtraction systems will adjust the parameters of their models to better match the existing environment. Simple background subtraction algorithms can be very fast, even when analyzing every pixel of an image. However, when using more advanced systems e.g., MGM, the processing of each single pixel becomes hazardous.

This is indeed a problem, since initially background subtraction systems gained popularity because they were much faster than other techniques like optic flow and template matching. The former technique tries to find the motion in an image sequence by matching certain features in the current frame to corresponsive features in the past frame. This way, if a spatial movement occurred, motion is sensed.[3] The latter technique uses templates of certain predefined objects (e.g., cardboard and stick models).[4] The images are then searched for good matches with these templates to facilitate object detection. Background subtraction is typically more common because of the lower computational cost compared to optical flow or template matching.

A common solution to the slow processing speed, is to decrease the frame rate at which the environment is monitored. Only regarding halve or one fourth increases the processing speed linearly. However, temporal consistency is very important to successfully track moving objects, so working with a reduced frame rate makes this difficult. During tracking, detected objects are matched with objects detected in previous frames to find the object trajectories. This matching also allows to identify the objects in newly captured frames.

A second approach is to reduce the image resolution. Evidently, when downscaling the images we get a linear increase in speed of the surveillance systems. In most cases the aspect ratio of the images is kept the same. Thus, downscaling the image results typically in an image which is one fourth of the original. If one changes the aspect ratio (e.g., by only analyzing the odd rows in an image) the objects become deformed, which can be bad for classification. One major drawback of downscaling this technique is that it introduces additional detection failures. Nowadays, captured surveillance images tend to be small. Large-scale surveillance systems work with multiple feeds, captured by a distributed camera system, and covering large sites. Therefore, to reduce bandwidth usage and processing times, low-cost cameras capture images in small resolutions (CIF to QCIF). Additionally, in the case of a surveillance camera covering wide ranges of an environment, specific objects of interest tend to be very small and only represent a couple of pixels of the entire frame. Therefore, simply downscaling the images will result in additional detection failures.

Finally, a popular method to increase the speed, is to apply course motion detection first before analyzing the entire frame.[5] In this case, simple frame differencing is used. When nothing happens in the scene, the frames are skipped; vice versa, if a large amount of motion is detected, the frame is analyzed. Although this works well in static controlled environments (e.g., indoor video surveillance), it would be bad in more complex environments (like outdoor scenes with movement of trees or crowded scenes) since movement will be detected everywhere. As such, the gain in speed is very dependent on the actual monitored environment and the activity within.

We present an analysis mask which identifies the pixels to be analyzed. The remaining pixels are processed by using a fast interpolation of the neighbouring pixels. As such, the input and output of our system is always the image at full size. Moreover, the analysis mask is made dynamic; it is reversed for each frame. This way, every pixel position in the image sequence is analyzed at certain times.

To evaluate our analysis mask, we apply it to the popular background subtraction technique MGM. Several researchers have based or compared their system on MGM. Wu et al. gave a concise overview of background subtraction algorithms, of which they have chosen MGM to compare with their own technique.[6] Additionally, they present a system which is better for localization and contour preserving, but it is more sensitive to complex environmental movements (such as waving trees). Since the background models in MGM have to be learned from observations of the scene, constructing a reliable model is hard when many moving objects are present. Lee et al. proposed an on-line expectation maximization learning algorithm for training adaptive Gaussian mixtures.[7] Their system allows to initialize the mixture models much faster than the original approach. Related to this topic, Zhang et al. presented an on-line background reconstruction method to cope with the initialization problem.[8]

It is generally known that MGM is a very popular approach for detection of moving objects. Since we do not change key components of MGM, we use this detection system for the evaluation of our analysis mask. Consequently, the mask could be combined with improved versions of MGM without decreasing the performance. In the next section we give a brief overview of MGM and show the complexity. Consequently, we elaborate on our analysis mask and show the results of the combination.

3. BACKGROUND SUBTRACTION USING MIXTURE OF GAUSSIAN MODELS

When using background subtraction, a background model is created that resembles the observed environment as closely as possible. In case of dynamic and complex environments, different background values for the same pixel are observed (e.g., movement of branches), so a single background model is insufficient. Moreover, environments tend to change over time. These changes which can be spread over long durations need to be adopted in the background model. Therefore, Stauffer and Grimson proposed MGM, which is a time-adaptive per pixel background subtraction technique that stores a number of models for each pixel.[2] Every pixel is represented by a vector, called I_p, consisting of three colour components (red, green, and blue). For each of these pixels a mixture of multivariate normal distributions, which are the actual models, is maintained and each of these models is assigned a weight.

$$G\left(I_p, \mu_p, \Sigma_p\right) = \frac{1}{\sqrt{(2\pi)^n \left|\Sigma_p\right|}} \, e^{-\frac{1}{2}\left(I_p-\mu_p\right)^T \Sigma_p^{-1}\left(I_p-\mu_p\right)}. \tag{1}$$

Equation (1) depicts a Gaussian distribution G. The parameters are μ_p and Σ_p, which are the mean and covariance matrix of the distribution respectively. For computational simplicity, the covariance matrix is assumed to be diagonal. For every new pixel a matching, an update, and a decision step are executed. The new pixel value is compared with the models of the mixture. A pixel is matched if its value occurs inside a confidence interval of 2.5 standard deviations from the mean of the model. In that case, the parameters of the corresponding distribution are updated according to equations (2), (3), and (4).

$$\mu_{p,t} = (1-\rho)\,\mu_{p,t-1} + \rho\left(I_{p,t}\right). \tag{2}$$

$$\Sigma_{p,t} = (1-\rho)\,\Sigma_{p,t-1} + \rho\left(I_{p,t}-\mu_{p,t}\right)\left(I_{p,t}-\mu_{p,t}\right)^T. \tag{3}$$

$$\rho = \alpha G\left(I_{p,t}, \mu_{p,t-1}, \Sigma_{p,t-1}\right). \tag{4}$$

The learning rate, α, is a global parameter and introduces a trade-off between fast adaptation and detection of slow moving objects. Each model has a weight, w, which is updated for every new image according to equation (5).

$$w_t = (1-\alpha)\,w_{t-1} + \alpha M_t \ . \tag{5}$$

If the corresponding model introduced a match, M_t is 1, otherwise it is 0. Equations (2) to (5) represent the update step. Finally, in the decision step, the models are sorted according to their weights. A threshold T is used to define which of the sorted models depict background or foreground. More specifically, the models for which the sum of their weights exceeds this threshold are regarded as background. MGM assumes that background pixels occur more frequently than actual foreground pixels. Indeed, if a pixel value occurs recurrently, the weight of the corresponding model increases and it is assumed to be background. If no match is found with the current pixel value, then the model with the lowest weight is discarded and replaced by a normal distribution with a

small weight, a mean equal to the current pixel value, and a large covariance. In this case the pixel is assumed to be a foreground pixel. MGM is different from conventional background subtraction systems in the sense that it maintains several models representing background and foreground. This allows to cope with the problem of moving objects which become static. New objects in an image will result in a new model in the mixture and if the object persists, the models' parameters will be fine-tuned and the weight increases. Eventually, the model will be regarded as background, so MGM can deal with parked cars, opened or closed doors, light switching, etc.

Figure 4, in section 5.2, shows the detection results of MGM for 2 different sequences. We have chosen a sequence recorded by an indoor and outdoor surveillance camera. The first sequence is part of the publicly available dataset from the Pets2001 workshop. It shows an outdoor scene with moving people and cars, moving trees and changing lighting conditions.[9] The second sequence was captured in a challenging indoor environment. This sequence is very noisy and the occurrence of shadows and reflections makes it difficult to reliably detect the moving objects. The first three columns of Figure 4 show the current frames, the actual ground truth and the results of MGM. No morphological post-processing has been done, so small misdetections are visible. The examples show that MGM does a good job in finding moving objects and dealing with the complex environments. However, MGM is not capable of dealing with shadows. Several techniques have been proposed to deal with this problem so we will not discuss this in our paper.[10, 11]

Since the process of matching, deciding and updating has to be done for each pixel in each frame, one sees that the complexity cannot be neglected. The processing speed of MGM will increase approximately linearly with the number of pixels to be analyzed. Therefore, in the next section, we propose an analysis mask that decides on which pixels will be fully processed. The other pixels will be interpolated from the detection result of the surrounding pixels.

4. PREPROCESSING

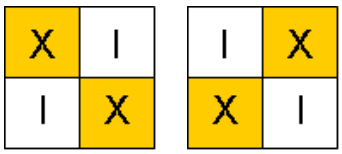

Figure 1. Left: 2x2 patch which constitutes the analysis mask for the odd frames, right: patch for the even frames.

To increase the processing speed we have chosen to use a checkerboard pattern for our analysis mask. For every frame we invert the mask so we get an odd and even analysis mask. These masks have the size of the entire frame and are constituted of 2x2 patches which are repeated over the frame. Figure 1 shows how the patches of odd and even analysis mask are constituted. The positions marked with an X denote pixels which are analyzed when the frame is processed. Alternatively, the positions marked with an I are interpolated from the surrounding pixels. Since the mask is switched for every frame, every pixel position in the surveilled content will be analysed once during two consecutive frames. This makes sure that small objects are not missed. Linear interpolation is used to predict the detection results of the pixels which were not analysed. To avoid storing the entire mask in memory, we can simulate the mask with following pseudo code:

```
for each pixel(x,y) in frame t
    if mod(x+y+t,2) == 1
      result(x,y) = doBackgroundSubtraction(pixel(x,y))
    else
      result(x,y) = interpolate(x,y)
```

Only half of the pixels are submitted to the background subtraction, in this case MGM. The other half is interpolated without updating the corresponding models. This means that the models are adapted based on only

half of the data that is collected over time. If we do not use the analysis mask the models would be updated for every new frame. However, our experiments, presented in the next section, show that this does not degrades the detection accuracy.

5. EXPERIMENTAL RESULTS

5.1. Execution Times

Table 1 shows a comparison of the execution times per frame for different test sequences. We have used the indoor and outdoor sequence presented in Figure 4 for our evaluation. The indoor sequence (IndoorGTTest2) shows fast moving people and the resolution is only 320 by 240 pixels. The outdoor sequence (PetsD2TeC2) covers a wide outdoor area. Consequently, the objects are longer visible and have sizes ranging from only a couple of pixels (people moving further away from the camera) to large blobs (nearby vehicles). The resolution of the latter sequence is 352 by 288 pixels (CIF). We assume that these sequences are representative for common surveillance video footage. For each frame the time it takes to perform the background subtraction (including matching, deciding, and updating) was recorded and consequently the average values and standard deviations are presented. As can be seen from the table, the proposed system outperforms MGM in processing speed for both the sequences. The right column shows the average reduction in processing speed per frame. The introduction of the analysis mask and the interpolation introduces a small delay so the gain in speed is slightly lower than 50%.

Table 1. Average execution times for MGM and the proposed system for the PetsD2TeC2 and the IndoorGTTest2 sequence.

	MGM		Proposed		
sequence	avg. (ms)	stdev. (ms)	avg. (ms)	stdev. (ms)	reduction (%)
PetsD2TeC2 (384x288)	97,3	1,4	56,3	1,4	42,1
IndoorGTTest2 (320x240)	62,3	0,7	33,4	0,7	46,4

5.2. Detection Results

To show the influence of the analysis mask on the performance of the detection, we present Receiver Operator Characteristic (ROC) graphs in Figure 2 and Figure 3. The figures show a quantitative comparison between MGM and our proposed system for the PetsD2TeC2 and the IndoorGTTest2 sequence. If a real background pixel is misclassified as foreground, it is called a false positive. If a foreground pixel is not detected, it is called a false negative. To find the false negatives and positives we have made a manual ground truth annotation for the sequences for every 50th frame. This ground truth was consequently compared with the output of the two algorithms. Finally, the total amount of false negatives and positives were counted for each sequence to create the ROC graphs. The X-axis shows the False Positive Rate (FPR) which defines how many incorrect positive results occur among all negative samples available during the test. In this case, it represents the number of pixels which were incorrectly considered as foreground, among all the real background pixels. The True Positive Rate (TPR), shown on the Y-axis, is the sensitivity and denotes the percentage of the real foreground pixels which were correctly classified. Good systems obtain a high TPR and low FPR. To get different values for the ROC curves, the learning speed α is variated from 0.0005 (very slow learning, the models are kept almost static) to 0.1 (very fast learning, new pixel values have much influence on the models). For each value of α the TPR and FPR is calculated and plotted. Values recorded with a high learning speed will typically be situated to the left on the curve. Indeed, if we have a high learning speed, the parameters and weights of the models are adapted more drastically and slow moving objects will be learned into the background faster, resulting in less true positives. When using a low learning speed, objects will not be learned into the background, but the models are not capable of adapting to small variations of the background. This results in more true positives but also in more false positives.

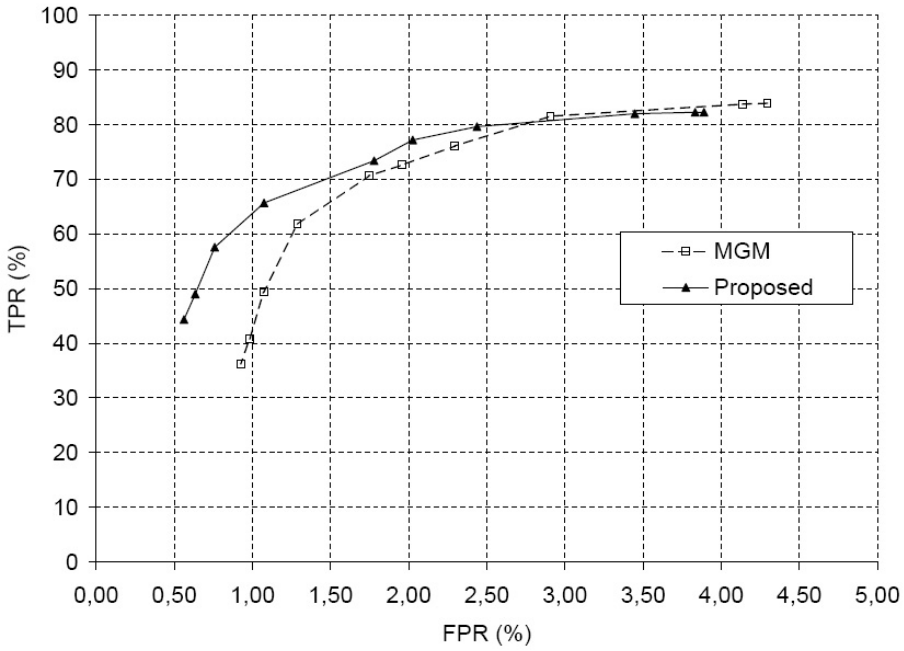

Figure 2. ROC graph for MGM and proposed on the "PetsD2TeC2" sequence

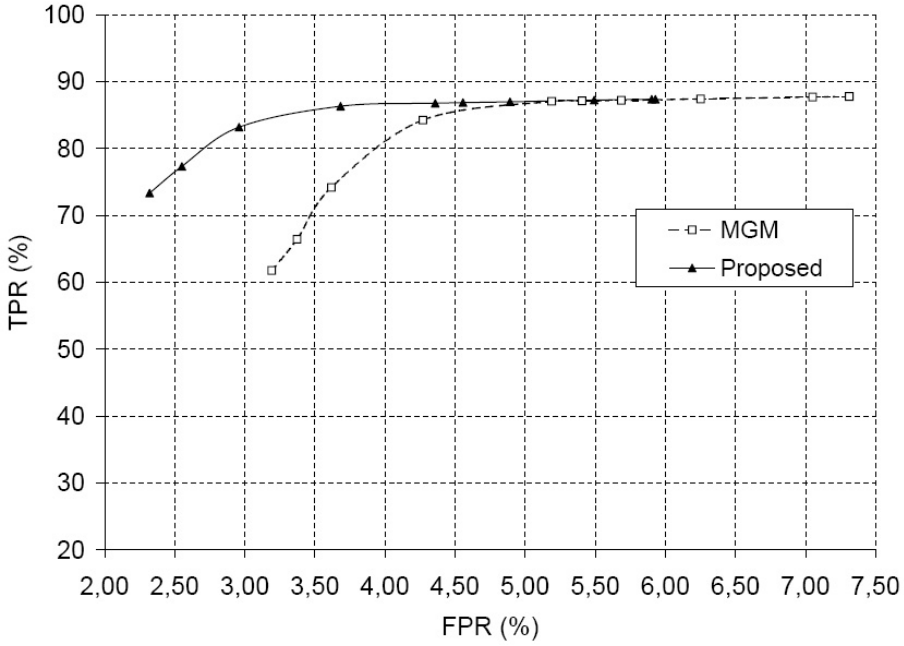

Figure 3. ROC graph for MGM and proposed on the "IndoorGTTest2" sequence

As can be seen in both figures, MGM and the proposed system obtain approximately the same TPR for a low learning speed so the detection of actual foreground regions (like people and vehicles) is similar. In fact, the proposed system performs even better for higher learning speeds. This is due to the interpolation and has two reasons. Firstly, when pixels of a real foreground object resemble the background, MGM will not be able to detect them. However, if surrounding pixels are detected as foreground pixels, the interpolation will see the

centre pixel as foreground too. Secondly, when (part of) objects are learned into the background due to the higher learning speed, this does not happen for all pixels of the object at once. Variations in the colour of the object itself causes some of the pixels to be merged faster into the background than others. When using the proposed system, during the time that an object is learned into the background, the interpolation causes some pixels to still be regarded as foreground.

The FPR for the proposed system is significantly lower than for MGM. In fact, the analysis mask acts as a morphologic filter since small noise is deleted in advance. As said before, we have not applied any morphological post-processing for either of the systems. If a connected component analysis would be done on the detection results of MGM, we would see that the ROC curves would lie closer together.

A visual comparison is given in Figure 4. The figure shows, from left to right, the current frame, the ground truth, the output of MGM and the output of the proposed system. It is clear that the moving objects in the scene are detected by both systems. We also can see the differences introduced by the interpolation. Small holes in the foreground object are filled, small noise in background regions is removed. This explains the ROC curves shown in Figure 2 and Figure 3.

6. CONCLUSIONS

This paper presented an analysis mask to reduce the number of pixels that need to be evaluated during object detection. The mask is a checkerboard pattern which is alternated for every frame. Only pixels denoted in this mask are analyzed and the remaining pixels are then interpolated from the surrounding pixels. To evaluate the system we have applied it to a popular background subtraction scheme, called the Mixture of Gaussian Models. We have shown the gains in processing speed on sequences with different characteristics. Additionally, we presented the detection results when using the normal background subtraction system and the system extended with the analysis mask. The results show that the use of the mask and the interpolation does not degrade the background subtraction. Moreover, the interpolation can act as a morphological filter and improves the outcome of the normal scheme.

ACKNOWLEDGMENTS

The research activities that have been described in this paper were funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT-Flanders), the Fund for Scientific Research-Flanders (FWO-Flanders), and the European Union.

REFERENCES

1. A. Dick and M. Brooks, "Issues in automated visual surveillance," in *Proceedings of International Conference on Digital Image Computing: Techniques and Applications*, pp. 195–204, 2003.
2. C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence* , pp. 747–757, 2000.
3. J. Barron, D. Fleet, and S. Beauchemin, "Performance of optical flow techniques," *International Journal of Computer Vision* , pp. 43–77, 1994.
4. H. Li, S. Lin, and Y. Zhang, "Combining template matching and model fitting for human body segmentation and tracking with applications to sports training," *Image Analysis and Recognition* , pp. 823–831, 2006.
5. P. Rosin and T. Ellis, "Image difference threshold strategies and shadow detection," in *Proceedings of the British conference on Machine vision*, pp. 347–356, 1995.
6. J. W. and M. Trivedi, "Performance characterization for gaussian mixture model based motion detection algorithms," in *Proceedings of the IEEE International Conference on Image Processing*, pp. 97–100, 2005.
7. D. Lee, "Online adaptive gaussian mixture learning for video applications," in *Lecture Notes in Computer Science, Statistical Methods in Video Processing*, pp. 105–116, 2004.
8. Y. Zhang, Z. Liang, Z. Hou, H. Wang, and M. Tan, "An adaptive mixture gaussian background model with online background reconstruction and adjustable foreground mergence time for motion segmentation," in *Proceedings of the IEEE International Conference on Industrial Technology*, pp. 23–27, 2005.

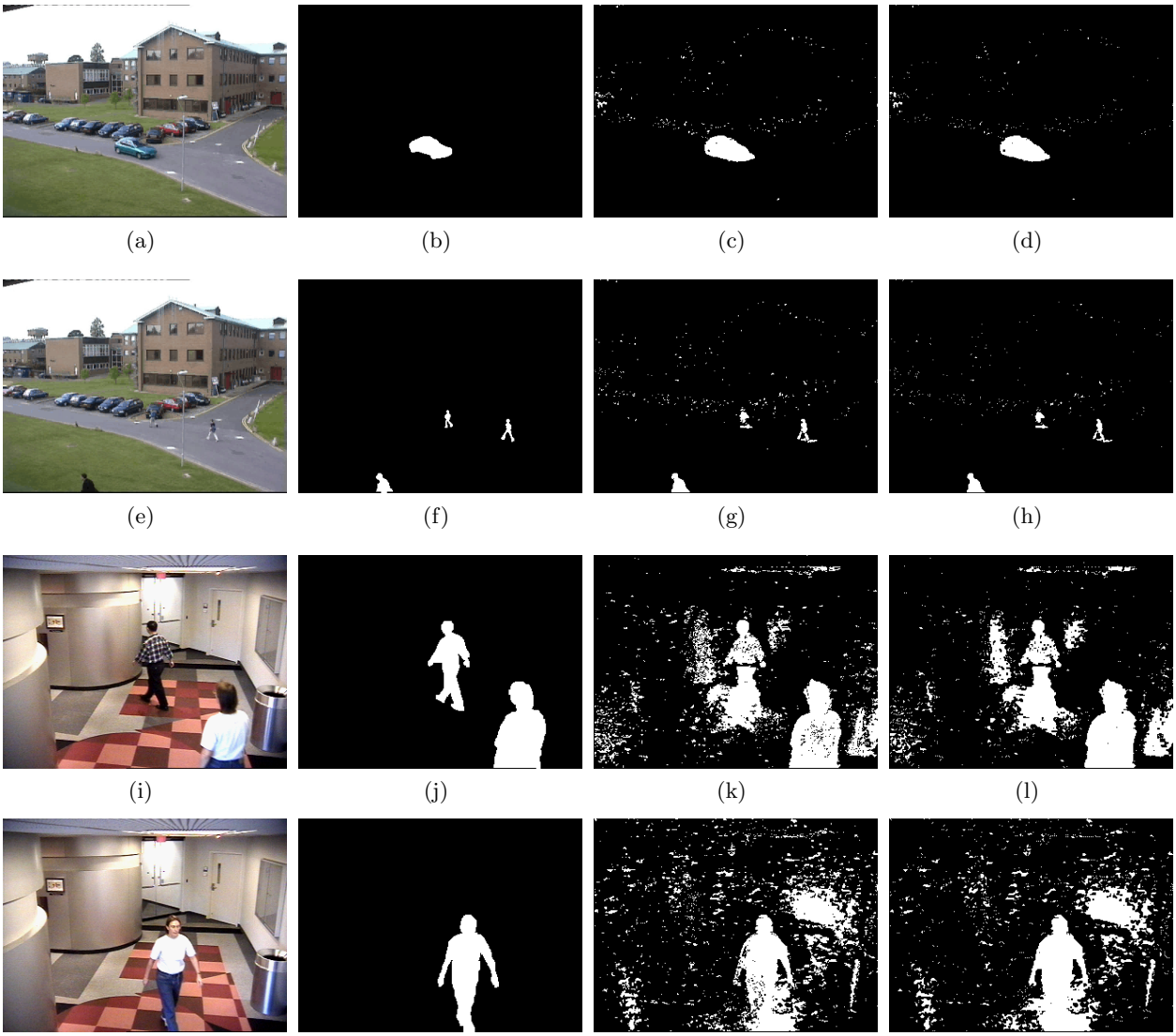

Figure 4. First column: current frame, second column: ground truth, third column: output by MGM, fourth column: output by Proposed. The first and second row shows the results for the PetsD2TeC2 sequence, frame 350 and 750 respectively. The third and fourth row shows the results for the IndoorGTTest2 sequence, frame 900 and 1100 respectively.

9. L. Brown, A. Senior, Y. Tian, J. Connell, A. Hampapur, C. Shu, H. Merkl, and M. lu, "Performance evaluation of surveillance systems under varying conditions," in *Proceedings of IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, pp. 23–27, 2005.
10. A. Pratie, I. Mikic, M. Trivedi, and R. Cucchiara, "Detecting moving shadows: Algorithms and evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence* , pp. 918–923, 2003.
11. R. Cucchiara, C. Grana, G. Neri, M. Piccardi, and A. Prati, "The sakbot system for moving object detection and tracking," *Video-Based Surveillance Systems - Computer Vision and Distributed Processing* , pp. 145–157, 1997.