A Perturbation Method for Evaluating Background Subtraction Algorithms

Thanarat Horprasert Chalidabhongse
Faculty of Information Technology
King Mongkut's Institute of Technology
Ladkrabang Bangkok 10520 Thailand
{thanarat@it.kmitl.ac.th}

Kyungnam Kim, David Harwood, Larry Davis
Computer Vision Lab, UMIACS
University of Maryland
College Park, MD 20742, USA
{knkim|harwood|lsd@umiacs.umd.edu}

Abstract

We introduce a performance evaluation methodology called Perturbation Detection Rate (PDR) analysis, for measuring performance of background subtraction (BGS) algorithms. It has some advantages over the commonly used Receiver Operation Characteristics (ROC) analysis. Specifically, it does not require foreground targets or knowledge of foreground distributions. It measures the sensitivity of a BGS algorithm in detecting low contrast targets against background as a function of contrast, also depending on how well the model captures mixed (moving) background events. We compare four algorithms having similarities and differences. Three are in [2, 3, 5] while the fourth is recently developed, called Codebook BGS. The latter algorithm quantizes sample background values at each pixel into codebooks which represent a compressed form of background model for a long image sequence.

1. Introduction

The capability of extracting moving objects from a video sequence captured using a static camera is a typical first step in visual surveillance. A common approach for discriminating moving objects from the background scene is detection by background subtraction. The idea of background subtraction is to subtract or difference the current image from a reference background model. The subtraction leaves only non-stationary or new objects.

Some background models assume that the series of intensity values on a pixel can be modeled by a single unimodal distribution. This basic model is used in [1, 2]. However, a single-mode model cannot handle multiple backgrounds, such as waving trees. The generalized mixture of Gaussians (MOG) has been used to model complex, non-static backgrounds [3, 4]. The MOG has some disadvantages. Background having fast variations cannot be accurately modeled with just a few Gaussians, causing problems for sensitive detection. To overcome these problems, a non-parametric technique [5] was developed for estimating background probabilities at each pixel from many recent samples over time using Kernel density estimation. It is able to adapt very quickly to changes in the background process.

The codebook (CB) background subtraction algorithm was intended to sample values over long times, without making parametric assumptions. It might be applicable to compressed video, which often has unusual, discontinuous distributions, as well as to uncompressed video. Mixed backgrounds can be modeled by multiple codewords, while brightness and color are separated.

When comparing BGS algorithms, ROC analysis is often employed when there are known background and foreground (target) distributions [8]. The ROC curves only measure the detection sensitivity for detecting a particular foreground against a particular background. There are as many ROC curves as there are possible different foreground targets. In addition, it will require considerable experimentation and ground-truth evaluation to obtain accurate false alarm rates (FA) and the miss detection rates (MD). However, in typical video surveillance applications, we usually are given a background scene for a fixed camera, but we do not or can not know what might possibly move in the scene as foreground objects. The perturbation method presented here, called perturbation detection rate (PDR) analysis, can measure the sensitivity of a BGS algorithm without assuming knowledge of the actual foreground distribution. Rather, it measures the detection of a variable, small ("just-noticeable") difference from the background, obtaining a foreground distribution by assuming that the foreground might have a distribution locally similar in form to the background, but shifted or perturbed. The detection is measured as a function of contrast, the magnitude of the shift or perturbation in uniform random directions in RGB.

In Section 2, we briefly describe the codebook-based method which is not given in the references. Then the performance evaluation technique, PDR analysis, is presented in Section 3 along with results for the present four algorithms. Conclusion and future work are given in Section 4.

2. Codebook-based background subtraction

The codebook BGS algorithm adopts a quantization/clustering technique, motivated by Kohonen [6, 7], to construct a background model from long observation sequences. For each pixel, it builds a codebook consisting of one or more codewords. Samples at each pixel are clustered into the set of codewords based on a color distortion metric together with a brightness ratio. Not all pixels have the same number of codewords. The clusters represented by codewords do not necessarily correspond to single Gaussian or other parametric distribution. Even if the distribution at a pixel were a single normal, there could be several codewords for that pixel. The background is encoded on a pixel by pixel basis. Thus a pixel is represented by a codebook which consists of one or multiple codewords.

Detection involves testing the difference of the current image from the background model with respect to color and brightness differences. Unlike MOG or the kernel methods [5], the codebook method does not involve floating point calculation of probabilities which can be costly. Indeed, the probability estimate in [5] is dominated by the nearby training samples. The CB method simply computes the distance of the sample from the nearest rescaled cluster mean. This is very fast and shows little difference in detection compared with the probability estimate. If an incoming pixel meets two conditions, it is classified as background - (1) The color distortion to some codeword is less than the detection threshold, and (2) its brightness lies within the brightness range of that codeword. Otherwise, it is classified as foreground. To cope with the problem of illumination changes such as shading and highlights, the CB method does not use RGB values directly. Brightness is often the largest source of variation, not intrinsic color. Physically these are different as well. The CB method calculates a brightness difference (a ratio of RGB absolute values) and a color difference which rescales codeword RGB values to the brightness of the current, tested pixel.

Figure 1 shows a typical application of the CB algorithm.

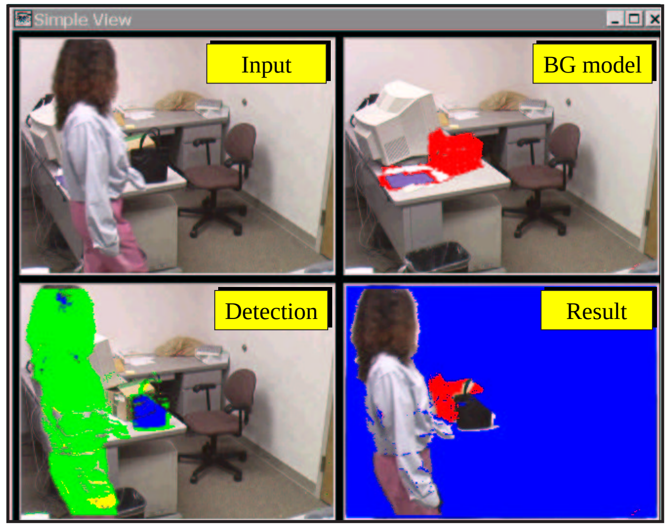


Figure 1: A background subtraction result obtained by the codebook-based method.

3. Performance evaluation

In this section, we propose a new methodology, called Perturbation Detection Rate (PDR) Analysis, for measuring performance of BGS algorithms, which is an alternative to the common method of ROC analysis. The purpose of PDR analysis is to measure the detection sensitivity of a BGS algorithm without assuming knowledge of the actual foreground distribution. The basic idea is to measure how far apart the two distributions must be in order to achieve a certain detection rate, or stated otherwise, given a false alarm rate (FA-rate), to determine detection rate as a function of the difference of the foreground from the background. It is similar to the *Just Noticeable Difference* (JND) typically used in comparing psychophysical magnitudes.

In general, detection accuracy depends on the algorithm and its parameters, shapes of the foreground and background distributions, and how far apart they are. In ROC, we assume we are given both foreground and background data of particular distribution shape and separation. We may vary the algorithm's parameters to obtain a certain combined false alarm rate and miss detection rate (or detection rate). Whereas, in PDR, we do not need to know exactly what the distributions are. The basic assumption made is that the shape of the foreground distribution is locally similar to that of the background distribution; however, foreground distribution of small (”just-noticeable”) contrast will be a shifted or perturbed version of the background distribution. This assumption is fairly reasonable because, in modeling video, any object with its color could be either background or foreground, e.g., a parked car could be considered as a background in some cases; in other cases, it could be considered a foreground target. Furthermore, by varying algorithm parameters we determine not a pair of error rates but a relation among the false alarm and detection rates and the distance between the distributions.

Given the parameters to achieve a certain fixed FA-rate, the analysis is performed by shifting or perturbing the entire BG distributions by vectors in uniformly random directions of RGB space with fixed magnitude Δ, computing an average detection rate as a function of contrast Δ. It amounts to simulating possible foregrounds at certain color distances. In the PDR curve, we plot the detection rate as a function of the perturbation magnitude Δ given a particular FA-rate.

First, we train each BGS algorithm on N training background frames, adjusting parameters as best we can to achieve a target FA-rate which would be practical in processing the video. Typically this will range from .01% to 1% depending on video image quality. To obtain a test foreground at color contrast Δ, we pass through the N background frames again. For each frame, we perturb a random sample of M pixel values (R_i, G_i, B_i) by a magnitude Δ in uniformly random directions.

The perturbed, foreground color vectors (R', G', B') are obtained by generating points randomly distributed on the color sphere with radius Δ. Then we test the BGS algorithms on these perturbed, foreground pixels and compute the detection rate for the Δ. By varying the foreground contrast Δ, we obtain an monotone increasing PDR graph of detection rates. In some cases, one algorithm will have a graph which dominates that of another algorithm for all Δ. In other cases, one algorithm may be more sensitive only in some ranges of Δ. Most algorithms perform very well for a large contrast Δ, so we are often concerned with small contrasts ($\Delta < 40$) where differences in detection rates may be large.

In this study, we compare four algorithms shown in Table 1. Since the algorithm in [5] accepts normalized colors (KER) or RGB colors (KER.RGB) as inputs, it has two separate graphs. Figure 2 shows the representative empty images from four test videos.

To generate PDR curves, we collected 100 empty consecutive frames from each video. 1000 points are randomly selected at each frame. That is, for each Δ, (100)$\times$(1000) perturbations and detection tests were performed. Those 100 empty frames are also used for training background models. During testing, no updating of the background model is allowed. For the non-parametric model in KER and KER.RGB, a sample size 50 was used to represent the background. The maximum number of Gaussians allowed in MOG is 4 for the video having stationary backgrounds and 10 for moving backgrounds. We do not use a fixed FA-rate for all four videos. The FA-rate for each video is determined by these three factors - video quality, whether it is indoor or outdoor, and good real foreground detection results for most algorithms. The FA-rate chosen this way is practically useful for each video. The threshold value for each algorithm has been set to produce a given FA-rate. In the case of MOG, the learning rate, α, was fixed to 0.01 and the minimum portion of the data for the background, T, was adjusted to give the desired FA-rate. Also, the cluster match test statistic was set to 2 standard deviations. Unless noted otherwise, the above settings are used for the PDR analysis.

Figures 4 and 5 show the PDR graphs for the videos in Figures 2(a) and 2(b) respectively.

For the indoor office video, consisting almost entirely of stationary backgrounds, CB and UNI perform better than the others. UNI, designed for unimodal backgrounds, has good sensitivity as expected. KER performs intermediately. MOG and KER.RGB do not perform as well for small contrast foreground Δ, probably because, unlike the other algorithms, they use original RGB variables and don't separately model brightness and color. MOG currently does not model covariances which are often large and caused by variation in brightness. It is probably best to explicitly model brightness. MOG's sensitivity is consistently poor in all our test videos, probably for this reason.

For the outdoor video, all algorithms perform somewhat worse even though the FA-rate has been increased to 1% from .01%. CB and KER, both of which model mixed backgrounds and separate color/brightness, are most sensitive, while, as expected, UNI does not perform well as in the indoor case. KER.RGB and MOG are also less sensitive outdoors, as before indoors.

Figure 3 depicts a real example of foreground detection, showing real differences in detection sensitivity for two algorithms. These real differences reflect performance shown in the PDR graph in Figure 6. The video image in Figure 3(a) shows someone with a red sweater standing in front of a brick wall of somewhat different reddish color. There are detection holes through the sweater (and face) in the MOG result (Figure 3(b)) . The CB result in Figure 3(c) is much better for this small contrast. After inspection of the image, the magnitude of contrast Δ was determined to be about 16 in missing spots. This was due to difference in color balance and not overall brightness. Figure 6 shows a large difference in detection for this contrast, as indicated by the vertical line.

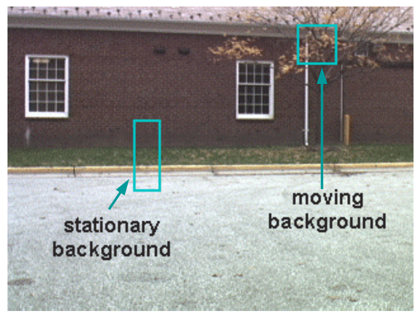
Figures 7, 8 and 9 show how sensitively the algorithms detect foregrounds against a scene containing moving backgrounds (trees) as well as stationary surfaces. In order to sample enough moving background events, 300 frames are allowed for training. As for previous videos, a PDR graph for the 'parking lot' video is given in Figure 7. Two windows are placed to represent 'stationary' and 'moving backgrounds' as shown in Figure 2(d). PDR analysis is performed on each window with the FA-rate obtained only within the window - a 'window' false alarm rate (instead of 'frame' false alarm rate).

Since most of the frame is stationary background, as expected, the PDR graph (Figure 8) for the stationary background window is very close to that for the entire frame. On the other hand, the PDR graph (Figure 9) for the moving background window is generally shifted right, indicating reduced sensitivity of all algorithms for moving backgrounds. Also, it shows differences in performance among algorithms, with CB and KER performing best. These results are qualitatively similar those for the earlier example of outdoor video shown in Figure 5. We can offer the

(a) indoor office

(b) outdoor woods

(c) red-brick wall



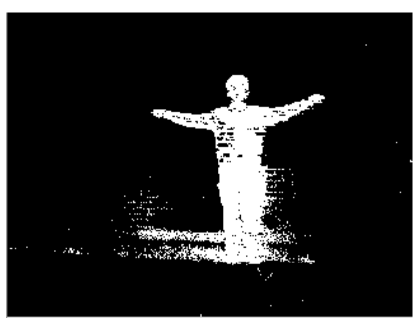
(d) parking lot

Figure 2: The sample empty-frames of four videos used in the experiments

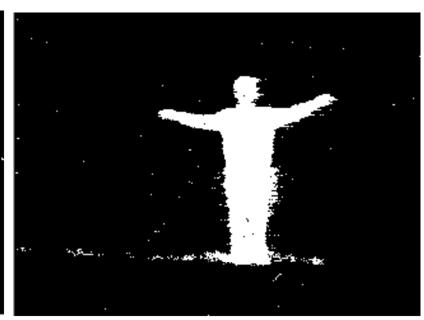
Name	Background subtraction algorithm
CB	codebook-based method described in Section 2
MOG	mixture of Gaussians described in [3]
KER and **KER.RGB**	non-parametric method using kernels described in [5]
UNI	unimodal background modeling described in [2]

Table 1: Four algorithms used in performance evaluation

(a) a 'red-brick wall' frame including a person in a red sweater



(b) detection using MOG



(c) detection using CB

Figure 3: Sensitive detection at small delta

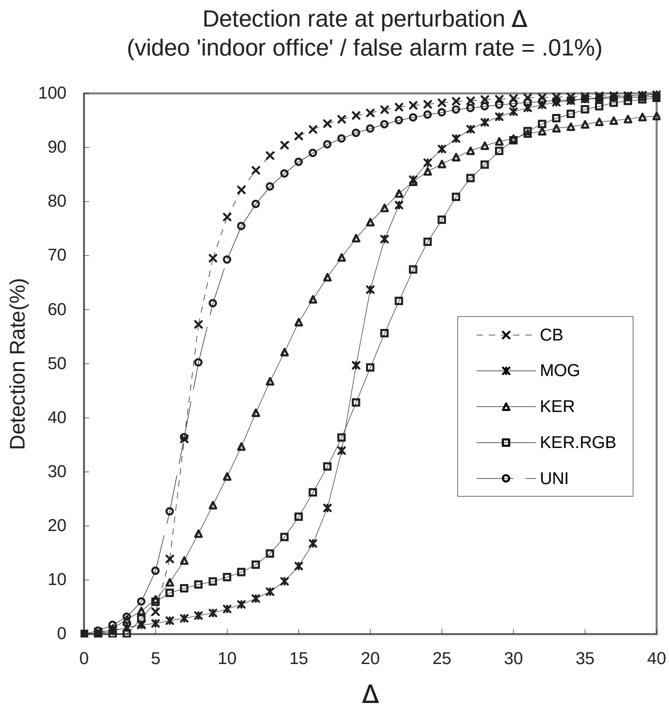


Figure 4: PDR for ’indoor office’ video in Figure 2(a)

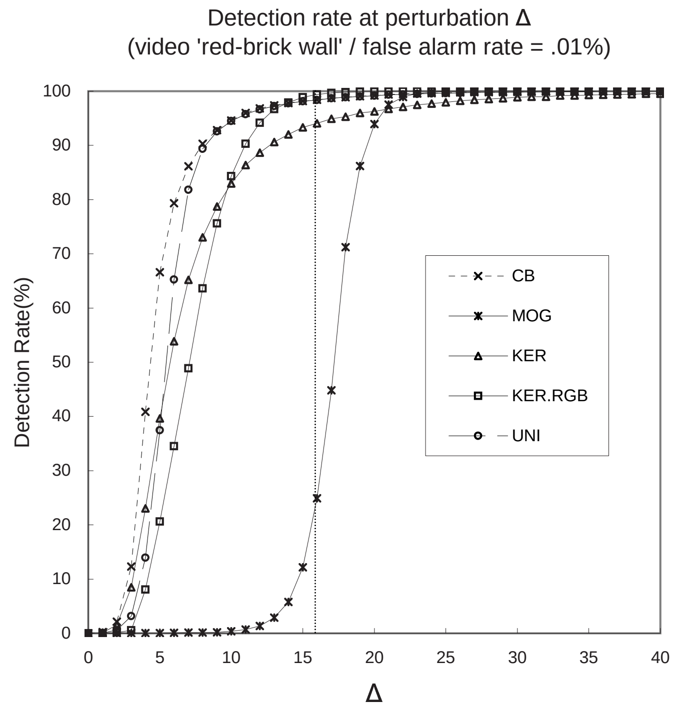


Figure 6: PDR for ‘red-brick wall’ video in Figure 2(c)

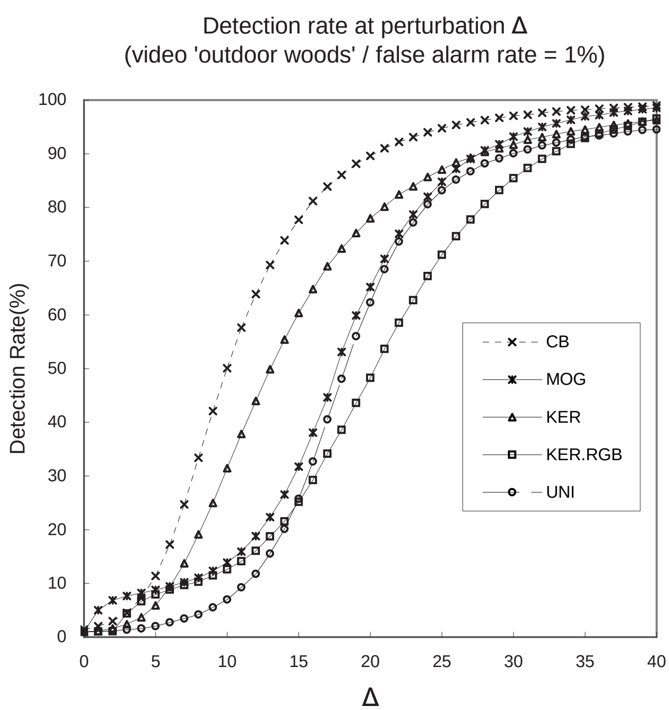


Figure 5: PDR for ’outdoor woods’ video in Figure 2(b)

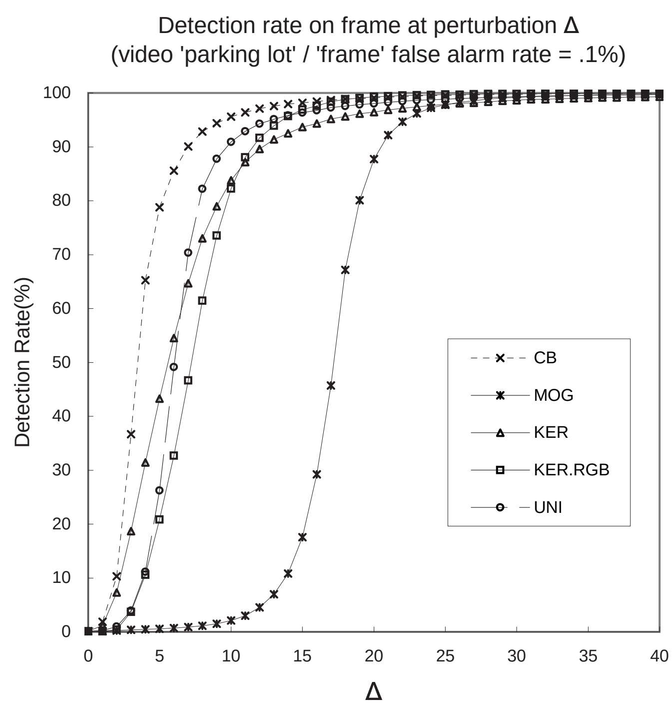


Figure 7: PDR for ‘parking lot’ video in Figure 2(d)

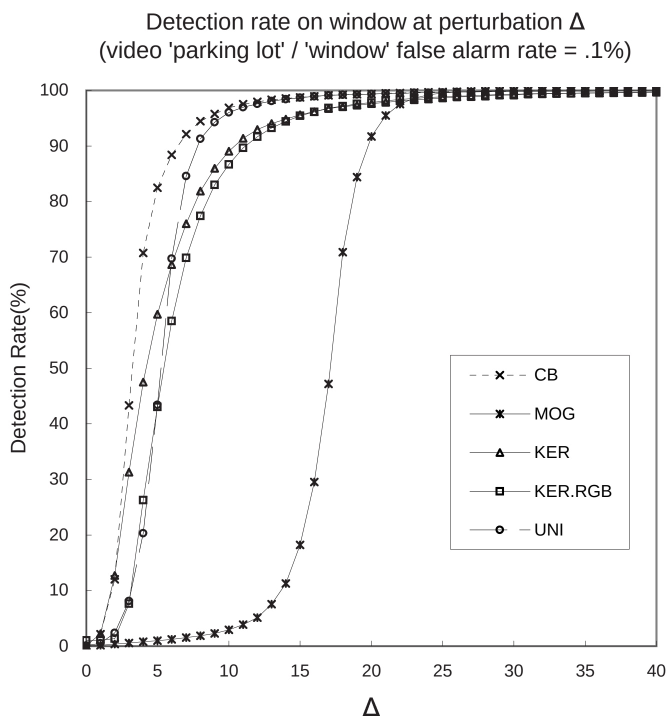


Figure 8: PDR for window on stationary background (Figure 2(d))

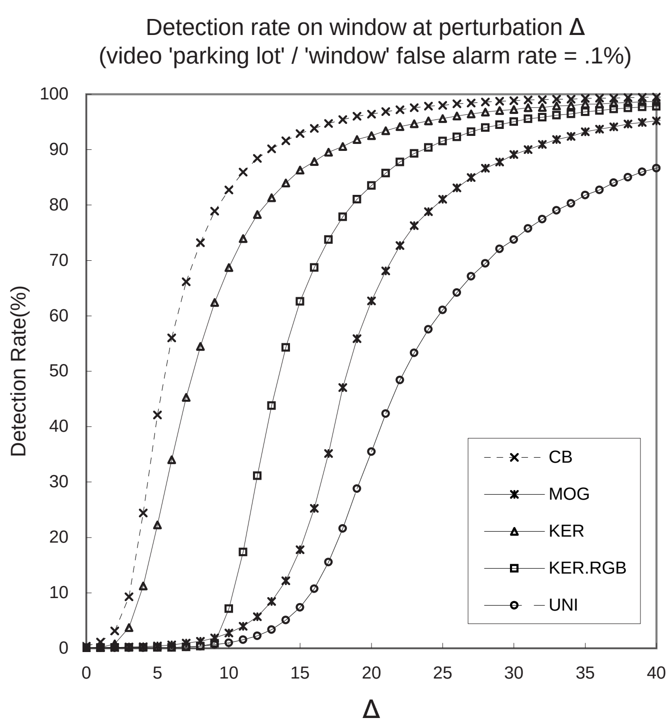


Figure 9: PDR for window on moving background (Figure 2(d))

same explanation as before: CB and KER were designed to handle mixed backgrounds, and they separately model brightness and color. In this video experiment, we had to increase the background sample size of KER (also that of KER.RGB) to 270 frames from 50 in order to achieve the target FA-rate in the case of the moving background window. It should be noted that CB, like MOG, usually models background events over a longer period than KER.

4. Conclusion and future work

After a brief description of the codebook-based BGS algorithm, we presented a perturbation method for measuring sensitivity of BGS algorithms. The PDR method does not require foreground targets in videos or knowledge of actual foreground distributions. PDR analysis does not consider all possible background or foreground distributions; it considers only those relevant to one video, scene and camera. It assumes that the foreground, when it has small contrast to the background locally, has a distribution similar in form to the background, but shifted or perturbed.

PDR analysis has two advantages over the commonly used ROC analysis: (1) It does not depend on knowing foreground distributions, (2) It does not need the presence of foreground targets in the video in order to perform the analysis, while this is required in the ROC analysis. Because of these considerations, PDR analysis provides practical general information about the sensitivity of algorithms applied to a given video scene over a range of parameters and FA-rates. In ROC curves, we obtain one detection rate for a particular FA-rate for a particular foreground and contrast.

We have applied the PDR analysis to four various BGS algorithms and four videos of different types of scenes. The results seem to be understandable, reflecting obvious differences among the algorithms as applied to the particular type of background scenes. We also provided a real video example of differences among the algorithms with respect to sensitive foreground detection which appears consistent with the PDR simulation.

There are limitations. The method doesn't model motion blur of moving foreground objects. Also in the case of mixed (moving) backgrounds, the simulated foreground distributions will be mixed (as plants or flags moving in the foreground); usually, though, foreground targets are from unimodal distributions. It should be noted, however, that the overall detection rates will be nearly the same if the clusters of the mixed distributions are well separated (compared to the usual small contrast delta). An important limitation is that foreground objects often will have shading and reflection effects on backgrounds, and these are ignored although they are important for choosing a proper, practical false alarm rate for real video analysis. (We have chosen practical false alarm rates for the videos used in this study.)

The present method would seem to be useful for qualitative comparison of sensitivity of different algorithms, as well as comparison of choice of parameters for a particular algorithm with respect to sensitivity. In the future, the present method could be extended to measure local detection rates throughout the frame of the scene or varying over time. This might have application to localized parameter estimation, e.g. of detection or adaptation parameters in different parts of the frame of the scene.

References

[1] C.R. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, no. 7, pp. 780-785, 1997.

[2] T. Horprasert, D. Harwood, and L.S. Davis, "A statistical approach for real-time robust background subtraction and shadow detection," *IEEE Frame-Rate Applications Workshop*, Kerkyra, Greece, 1999.

[3] C. Stauffer and W.E.L. Grimson, "Adaptive background mixture models for real-time tracking," *Int. Conf. Computer Vision and Pattern Recognition*, Vol. 2, pp. 246-252, 1999.

[4] M. Harville, "A framework for high-level feedback to adaptive, per-pixel, mixture-of-gaussian background models," *European Conf. Computer Vision*, Vol. 3, pp. 543-560, 2002.

[5] A. Elgammal, D. Harwood, and L.S. Davis, "Non-parametric model for background subtraction," *European Conf. Computer Vision*, Vol. 2, pp. 751-767, 2000.

[6] T. Kohonen, "Learning vector quantization," *Neural Networks*, Vol. 1, pp. 3-16. 1988.

[7] B.D. Ripley, "Pattern Recognition and Neural Networks," *Cambridge University Press*, 1996.

[8] X. Gao, T.E. Boult, F. Coetzee, V. Ramesh, "Error Analysis of Background Adaption," *Proc. Int. Conf. Computer Vision and Pattern Recognition*, pp. 503-510, 2000.