

# Evaluating Motion Detection Algorithms: Issues and Results

J. Renno, N. Lazarevic-McManus, D. Makris and G.A. Jones

Digital Imaging Research Centre, Kingston University,  
Penrhyn Road, Kingston upon Thames, Surrey, UK KT1 2EE  
[www.kingston.ac.uk/dirc](http://www.kingston.ac.uk/dirc)

## Abstract

*Motion detection is a fundamental processing step in the majority of visual surveillance algorithms. While an increasing number of authors are beginning to perform quantitative comparison of their algorithms, most do not address the complexity and range of the issues which underpin the design of good evaluation methodology. In this paper we present a motion detection algorithm with a number of novel contributions one of which specifically addresses the problem of large and sudden lighting variations caused by sunlight. A motivated and comprehensive comparative evaluation methodology is described and used to compare our proposed motion detection algorithm to two well-known techniques reported in the literature.*

## 1. Introduction

Motion detection is a fundamental processing step in the majority of visual surveillance algorithms. While an increasing number of authors are beginning to perform quantitative comparison of their algorithms, most do not address the complexity and range of the issues which underpin a good evaluation methodology. Such issues include the distinction and relative merits of pixel-based versus object-based metrics; the motivation of appropriate metrics; the impact of defining the end application; making explicit evaluation parameters and selecting appropriate values; and the role of ROC optimisation of algorithms. In this paper we review the performance evaluation literature, discuss some of the more complex issues, and propose a motivated and comprehensive comparative evaluation methodology based on ROC optimisation and a proposal for standardised end-user applications as context.

Sudden lighting conditions are particularly problematic for motion detection algorithms. Compounded by the compensating response taken by most cameras, the result is a significant change in both intensity and colour. We introduce a novel technique for handling such rapid lighting variations based on the observation that these global changes give rise to correlated changes in UV. A search region for constraining these colour changes is controlled by the global *mean colour difference* of the frame.

ROC analysis is used to optimise the selection of parameters and to verify the effectiveness of a novel area thresholding process. The proposed motion detection algorithm is

comparative evaluated against two well-known techniques reported in the literature.

## 2. Previous Work

A number of techniques have been proposed dedicated to performance analysis of visual surveillance algorithms. Many of them deal with evaluation of detection of moving objects [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12], whereas others address evaluation of the tracking of detected objects throughout the video sequence or both [13, 14, 15, 16, 17, 18]. Since successful tracking relies heavily on accurate object detection, the evaluation of object detection algorithms within a surveillance systems plays an important part in overall performance analysis of the whole system.

### Ground Truth

Evaluation based on GT offers a framework for objective comparison of performance of alternate surveillance algorithms. Such evaluation techniques compare the output of the algorithm with the GT obtained manually by drawing bounding boxes around objects, or marking-up the pixel boundary of objects, or labelling objects of interest in the original video sequence. Manual generation of GT is an extraordinarily time-consuming and tedious task and, thus, inevitably error prone even for motivated researchers. (See List *et al* [19] for an interesting study on inter-observer variability in this context.) Black *et al* recently proposed the use of a semi-synthetic GT where previously segmented people or vehicles are inserted into real video sequences [13].

Interpretation of evaluation results is obviously based on the type of GT used for comparison. However, established standards for GT are only just emerging. There are several ambiguities involved in the process of GT generation. For example, whether to account only for individual objects or also for groups of objects, or whether to look at the bounding boxes or exact shapes of objects. Several GT generation tool are available: ViPER [5], ODViS [16], CAVIAR [20]. Standardisation of datasets has been championed by PETS<sup>1</sup>. Nationally funded initiatives currently preparing datasets include the French *ETISEO* project<sup>2</sup> and the UK Home Office *iLIDS* project<sup>3</sup>.

<sup>1</sup><http://www.cvg.cs.rdg.ac.uk/cgi-bin/PETSMETRICS/page.cgi?dataset>

<sup>2</sup>[www.silogic.fr/etiseo/](http://www.silogic.fr/etiseo/)

<sup>3</sup><http://scienceandresearch.homeoffice.gov.uk/hosdb/news-events/270405>

## Common Performance Metrics

Performance evaluation algorithms based on comparison with ground truth can be further classified according to the type of metrics they propose. Typically, ground-truth based metrics are computed from the *true positives* (TP), *false positives* (FP), *false negatives* (FN), and *true negatives* (TN), as represented in the *contingency table* below. For *pixel-based* metrics FP and FN refer to pixels misclas-

Output Class	True Class	
	Foreground	Background
Fore	True Positives (TP)	False Positives (FP)
Back	False Negatives (FN)	True Negatives (TN)

Table 1: Contingency Table

sified as foreground (FP) or background (FN) while TP and TN account for accurately classified pixels[2, 3, 4, 7, 14, 17, 13]. Usually, they are calculated for each frame and an overall evaluation metric is found as their average over the entire video sequence. For object-based metrics TP refers to the number of detected objects sufficiently overlapped by GT, FP to the number of detected objects not sufficiently overlapped by the GT, and FN to the number of GT objects not sufficiently covered by any automatically detected objects[6, 15, 11]. (Note that this *degree of overlap* is a parameter of the evaluation process.) Some authors combine both types[5]. Furthermore, a number of methods evaluate individual objects by weighting misclassified pixels according to their impact on the quality of segmented object[1, 8, 9, 10, 12] - in essence, pixel-based methods.

Typical metrics computed per-frame or per-sequence are the *true positive rate* (or *detection rate*)  $t_p$ , *false positive rate*  $f_p$ , *false alarm rate*  $f_a$  and *specificity*  $s$

$$t_p = \frac{N_{TP}}{N_{TP} + N_{FN}}, \quad f_p = \frac{N_{FP}}{N_{FP} + N_{TN}}, \quad (1)$$

$$f_a = \frac{N_{FP}}{N_{TP} + N_{FP}}, \quad s_p = \frac{N_{TN}}{N_{FP} + N_{TN}} \quad (2)$$

where  $N_{TP}$ ,  $N_{FP}$ ,  $N_{TN}$  and  $N_{FN}$  are the number of pixels or objects identified as *true positives*, *false positives*, *true negatives* and *false negatives* respectively.

In some applications (*e.g.* facial identification) competing algorithms are presented with images which contain known *clients* and *imposters*. For object-based motion detection evaluation, there is no equivalent prior set of known *imposters* i.e. false objects in the ground truth! Thus, as it is not possible to identify *true negatives*, the *false positive rate* cannot be computed.

The great majority of proposed metrics are restricted to pixel-level discrepancy between the detected foreground and the ground-truth - namely false positive and false negative pixels. These metrics are useful to assess overall segmentation quality on a frame-by-frame basis but fail to provide an evaluation of individual object segmentation. Often these measures are normalised by image size or the amount of detected change in the mask[10], or object *relevance*[1].

However, the more principled approach is based on Receiver Operating Curves (ROCs).

Evolved to characterise the behaviour of binary classifiers, ROC curves plot *true positive rate* against *false positive rate* to facilitate the selection of optimal classification parameters and compare alternative classification techniques[3, 6]. An ROC graph is an alternative presentation to plotting metrics for each frame in the sequence which is often difficult to assess by a reader[9].

## Other Performance Metrics

All pixel-based methods which evaluate individual object segmentation rely on existence of shape-based ground-truth mask generated by costly process if they are to avoid errors. In addition to the advantage of avoiding hand labelling individual foreground pixels in every frame, object-based methods only require ground-truth in the form of bounding-boxes[6, 15, 11]. The object-level evaluation proposed by Hall et al[11] plots detection rates and false alarm rates using various values of overlap threshold to determine association with the GT. As they do not have true-negatives, false alarm rate is computed as alternative to false positive rate and the area under the curve is used as a measure of performance. Other object-based metrics proposed are based on the similarity of detected and ground-truth objects *i.e.* relative position[16, 11] or shape, statistical similarity and size[1, 11].

A major problem in motion detection is the fragmentation and merging of foreground objects. While these will impact on pixel-based metrics, a number of explicit metrics have been proposed[6, 5]. Typically these measure the average number of detected regions overlapping each ground-truth object and average number of ground-truth objects associated with multiple detected regions.

Metrics may also be designed to take account of human perception of error where false positives and false negatives hold different levels of significance by introducing weighting functions for misclassified pixels on an object-by-object basis[8, 10]. Villegas and Marichal[8] increase the influence of misclassified pixels further from the boundary of ground-truth objects. Cavallaro *et al*[10] account for temporal effects of *surprise* and *fatigue* effects where sudden changes in quality of segmentation amplifies error perception.

## 3. Motion Detection Algorithm

Motion detection algorithms aim to detect moving objects whilst suppressing false positives caused by lighting changes, moving background, shadows and *ghosts*. We introduce a novel technique for handling rapid lighting conditions (that normally result in false detection) using the observation that these global changes give rise to correlated changes in UV. In addition, two Gaussian colour probability density functions (PDF) are used to model different aspects of a background pixel. Using these volumes it is possible to classify each new pixel as *foreground*, *shadow*, *highlight* or

*background*. To validate the approach, the method is compared against other published methods[21, 22].

### 3.1. Pixel Classification

The classification of each pixel into the three *background*, *shadow* or *foreground* labels proceeds as follows:

**Background:** A three-dimensional Gaussian PDF in YUV colour-space is used to model the appearance of the background. Each pixel is assigned a *Background* label if it lies within a *background volume*  $\mathcal{B}^{YUV}$  defined by the Chi-squared distance  $\chi_{\tau_B}^2$  from the PDF mean. The radius is determined by the classification parameter  $\tau_B$ ;  $\{0 \leq \tau_B < 1\}$  which determines the proportion of the distribution contained in the volume.

**Global Lighting Change:** For pixels not labelled as *Background*, further classification is undertaken to detect background pixels affected by *global light changes*. To achieve this we shall assume that as global light intensity levels across the image increase and decrease, there is usually a correlated increase or decrease in the UV content at each pixel. To capture the impact of these global intensity changes on a pixel, we define a dynamic rectangular region  $\mathcal{R}^{UV}$  in  $U, V$  (see Figure 1) whose position and orientation are determined by the mean of the background PDF, and whose dimensions  $L, W$  are determined by the *mean colour difference*  $D$  of the whole frame. To retain a background label,  $U, V$  must be located within  $\mathcal{R}^{UV}$ .

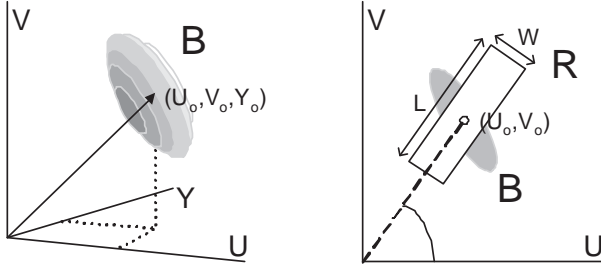


Figure 1: Pixel Labelling

If  $\bar{U}$  and  $\bar{V}$  are the average  $U, V$  components in a frame, then *mean colour difference* is defined as  $D = |\bar{U} - \bar{V}|$ . The minimum and maximum mean colour differences over a typical day,  $D_{\min}$  and  $D_{\max}$  respectively, are usually computed at setup. The mean of  $\mathcal{R}^{UV}$  is  $U_0, V_0$ , and its orientation is defined by a rotation  $\theta$  from the  $U$ -axis to the line connecting the mean to the origin *i.e.*  $\tan \theta = V_0/U_0$ . Length  $L$  is defined as  $L = D - D_{\min}$  while the width is currently 20% of  $L$ .

**Shadow:** A common approach to locating shadows is to assume that any significant intensity change without significant chromaticity change has been caused by shadow[23]. Chromaticity is computed using *normalised RGB* (nRGB) colour space, and as before, each pixel is modelled by a 3D Gaussian PDF. Pixels are classified as shadow if they lie within a *Shadow* volume  $\mathcal{S}^{nRGB}$  whose boundary is defined by the Chi-squared distance  $\chi_{\tau_S}^2$  from the PDF mean. The

radius is determined by the second classification parameter  $\tau_S$  which determines the proportion of the distribution contained in the volume  $\{0 \leq \tau_S < 1\}$ .

Having defined the three decision spaces  $\mathcal{B}^{YUV}$ ,  $\mathcal{R}^{UV}$  and  $\mathcal{S}^{nRGB}$ , we can assign each pixel  $\phi$  a label  $\lambda_\phi$  as follows

$$\lambda_\phi = \begin{cases} \text{Background} & \text{if } (\phi \in \mathcal{B}^{YUV}) \text{ or } (\phi \in \mathcal{R}^{UV}), \\ \text{Shadow} & \text{elseif } \phi \in \mathcal{S}^{nRGB}, \\ \text{Foreground} & \text{else.} \end{cases}$$

### 3.2. Updating Colour PDFs

A Gaussian PDF is represented by its mean  $\mu$  and scatter matrix  $\Sigma$  (a precursor of the covariance). The usual (and simplest) approach to updating the Gaussian PDFs of background pixels is to employ a temporal weighting scheme based on the current values *i.e.*

$$\begin{aligned} \mu_t^{YUV} &= \alpha_\mu \mathbf{c}_t^{YUV} + (1 - \alpha_\mu) \mu_{t-1}^{YUV} \\ \Sigma_t^{YUV} &= \alpha_\Sigma (\mathbf{c}_t^{YUV})(\mathbf{c}_t^{YUV})^T + (1 - \alpha_\Sigma) \Sigma_{t-1}^{YUV} \\ \mu_t^{nRGB} &= \alpha_\mu \mathbf{c}_t^{nRGB} + (1 - \alpha_\mu) \mu_{t-1}^{nRGB} \\ \Sigma_t^{nRGB} &= \alpha_\Sigma (\mathbf{c}_t^{nRGB})(\mathbf{c}_t^{nRGB})^T + (1 - \alpha_\Sigma) \Sigma_{t-1}^{nRGB} \end{aligned}$$

where  $\mathbf{c}^{YUV}$  and  $\mathbf{c}^{nRGB}$  are column vectors containing the colour components of a pixel, and  $\alpha_\mu, \alpha_\Sigma$  are the update weights for the mean and scatter respectively. (Typically to ameliorate noise,  $\alpha_\Sigma \approx \alpha_\mu^2$ .) In practice, by assuming that the components of the colour spaces are independent, considerable reduction in processing time (for updating scatter matrix, inverting covariances and computing Mahalanobis distances) can be achieved.

In the above **Fixed** strategy, the update factors are applied even if the pixel is labelled as foreground. In practice suspending updating when labelled as foreground is problematic; leading to *lock-out* effects when previous stationary parts of the scene start moving. Instead we shall investigate an **Adaptive** update strategy using a different lower update weight for foreground pixels *i.e.*

$$\alpha_\mu = \begin{cases} \alpha_\mu^{\text{slow}} & \text{if } \lambda = \text{Foreground}, \\ \alpha_\mu & \text{else.} \end{cases} \quad (3)$$

### 3.3. Area Thresholding

A connected-components algorithm is now applied to the detection mask to eliminate small regions using a constant threshold  $\tau_A$ . In this **Constant** strategy, a fixed area threshold will disadvantage small distant objects. A second **Linear** variant of this algorithm is employed which linearly scales the threshold as a function of the vertical distance from the horizon. (Obviously we assume that the horizon  $i_h$  is recovered at setup[24].) Thus, for an object whose base is located at pixel row  $i$

$$\tau_A = \gamma_A (i_h - i) + \tau_{A_{\min}} \quad (4)$$

where  $\gamma_A$  and  $\tau_{A_{\min}}$  are algorithm parameters.



Figure 2: Evaluation Dataset

## 4. Evaluation Methodology

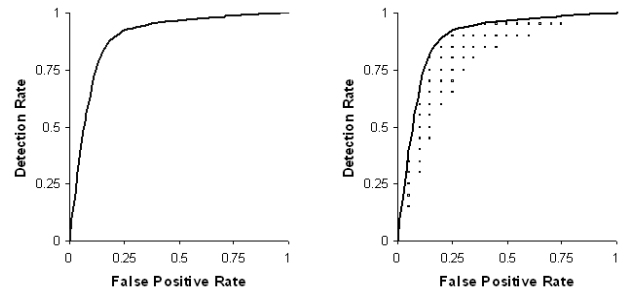
### 4.1. Dataset and Ground Truth

The dataset consists of 8210 frames recording activities in a car park at full frame-rate covering a period of five and a half minutes. (Example frames shown in Figure 2.) The CCTV camera has iris auto-correction and colour switched on. The video sequence includes a total of 24 moving objects, people and vehicles appearing at close, medium and far distances from the camera. There are a variety of both gradual and sudden lighting changes present in the scene due to English weather conditions (bright sunshine interrupted by fast moving clouds, reflections from windows of vehicles and buildings). There are both static and dynamic occlusions present in the scene with moving objects crossing paths and disappearing partly or totally behind static objects. In addition, a strong wind causes swaying trees and bushes to disturb the background.

The ground truth is generated manually (by one person) using an in-house semi-automatic tool for drawing bounding boxes for every target within each frame of the video sequence. Ground truth provides the temporal ID of the object, its bounding box enclosing pixels of interest, defines the type of the object whether person or vehicle, and defines the degree of occlusion with other objects *i.e.* unoccluded, semi-occluded or fully-occluded.

### 4.2. ROC-based Analysis

Receiver Operating Curves (ROC) are a useful method of interpreting performance of a binary classifier. ROC curves graphically interpret the performance of the decision-making algorithm with regard to the decision parameter by plotting the *True Positive Rate* ( $t_p$ ) against the *False Positive Rate* ( $f_p$ ). Each point on the curve is generated for the range of decision parameter values - see Figure 3(a). In foreground detection, such decision parameters could be a threshold on the greylevel difference between incoming pixel and reference pixel, or a threshold on the size of any foreground object. When there is more than one classification parameter, a distribution of points representing all parameter value combinations is generated in the ROC space. The required ROC curve is the top-left boundary of the convex hull of this distribution as shown in Figure 3(b).



(a) Single Classification Parameter (b) Multiple Parameters

Figure 3: Generating ROC Curves

In general the optimal operating point is chosen in the top left quadrant of the ROC space and is defined as the classification parameter value on the iso-performance line with the lowest misclassification cost. The gradient  $\lambda$  of this line is defined as

$$\lambda = \frac{(1 - P_T) C_{FP}}{P_T C_{FN}} \quad (5)$$

where  $P_T$  is the prior probability of a foreground pixel (or object), and  $C_{FN}$  and  $C_{FP}$  are the cost of classifying a moving pixel (or object) as stationary and vice versa. Misclassification costs depends on the intended application of motion detection (*e.g.* tracking, counting people, alarming, detecting a specific person, *etc*) and the ratio of foreground pixels (or objects) to background in the GT. Points on the graph lying above-left of the line have a lower misclassification cost while points below-right of the line have larger costs. The misclassification cost  $C$  at the operating point  $t_p^*, f_p^*$  is given by

$$C(t_p^*, f_p^*) = (1 - P_T)C_{FP}f_p^* + P_T C_{FN}(1 - t_p^*) \quad (6)$$

### Cost Scenarios

To explore the effect of the cost ratio, we shall introduce two different cost scenarios: the *Ticket Fraud* scenario in which, say, the cost of **detaining** an innocent member of the public  $C_{FP}$  is defined as double the cost of failing to catch a ticket fraudster  $C_{FN}$ ; and the *Evidence Gathering* scenario in which the cost of **video-ing** an innocent passerby  $C_{FP}$  is, say, 10 times smaller than the cost of failing to video a

terrorist bomber  $C_{FN}$ <sup>4</sup>. The relative costs are arbitrary, and the relationship of these applications to motion detection is indirect. However, these different cost ratio scenarios ensure we are mindful of the ultimate application in the evaluation stage. (Obviously defining the social costs of violations of *libertarian* and *public safety* concepts is extremely fraught!)

### Evaluation Parameters

Typical performance evaluation takes the output of some visual surveillance algorithm and compares it with *ground truth* data - as illustrated in Figure 4. The performance of any surveillance algorithm will depend on the choice of the internal parameters. Optimal performance is typically determined using the ROC methodology discussed above.

Crucially, however, the result will also depend on the inevitable array of parameters required by the evaluation module *e.g.* the degree of overlap between the detected moving regions and the ground truth objects. How would you select appropriate values for these? A naive approach would be to include these evaluation parameters within the ROC methodology to select the optimal algorithm **and** evaluation parameter values. However, the result would be to evaluate each alternative surveillance algorithm with a **different** evaluation algorithm. Hardly an objective comparison! In our case, we propose to determine appropriate evaluation parameters using the ROC analysis applied to a competitor algorithm.

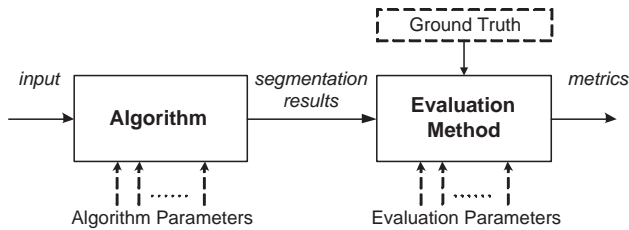


Figure 4: Typical performance evaluation system

### 4.3. Proposed Metrics

In pixel-based performance analysis, the label of each pixel (TP, FP, FN, and TN) is defined as follows: detected foreground pixels inside the GT bounding box (TP), detected foreground pixels outside the GT bounding box (FP), detected background pixels inside the GT bounding box (FN) and detected background pixels outside the GT bounding box (TN). As the ground-truth is only specified to the level of the bounding box, true-positive rates do not reach 100%.

**ROC Analysis and Classification Costs:** ROC analysis will be performed for each presented algorithm to identify optimal parameters (irrespective of published values) for the proposed *Ticket Fraud* and *Evidence Gathering* scenarios. The *Classification Cost* (Cost) at these operating points for each scenario will be recorded per algorithm.

**Signal Rate:** The signal rate  $s_r$  provides an insight into the localised quality of segmented objects, and is defined as

$$s_r = \frac{N_{TP}}{N_{TP} + N_{FP}^*} \quad (7)$$

where  $N_{TP}$  is the number of detected pixels within the bounding boxes of the ground truth, and  $N_{FP}^*$  represents any false positives that have pixel connectivity to the ground-truth. The behaviour of this metric is illustrated in Figure 5. Rather than measure the global signal confusion, this version attempts to measure the degree of local confusion surrounding detected objects.



Figure 5: SNR Metric: (Left) Original with GT, (Middle) Detected Blobs (Right) Connected false positives

**Specificity:** During sudden lighting changes, the detection method should ideally avoid classifying large parts of the image as foreground. Many metrics do not necessarily capture this behaviour as the number of true positive pixels paradoxically increases. The *Specificity* metric (see equation 2) essentially measures the number of background pixels which remain background, and hence drops dramatically if the detection method fails to cope with light changes.

To motivate the selection of appropriate object-based metrics, we note that motion detection is usually followed by a blob tracker to establish the temporal history of any detected object. The performance of this tracker will depend crucially on (i) the proportion of true objects located, (ii) the degree of local confusion caused by falsely detected objects, and (iii) the degree of fragmentation of true objects.

Object-based metrics pose a problem which does not arise for pixel-based metrics: establishing the correspondence between GT objects and the inevitably fragmented, merged and displaced detected blobs - particularly problematic as the density of objects rises and in the presence of noise objects. Following a typical approach, we use the degree of overlap between detected objects and GT bounding boxes to establish this correspondence. In general, this can result in one-to-many and many-to-one relationships. Thus the number of true positions  $N_{TP}$  and false negatives  $N_{FN}$  can be larger than the number of ground truth objects  $N$  *i.e.*  $N_{TP} + N_{FN} \geq N$ .

In object-based performance analysis, the label of each object (TP, FP, and FN) is defined as follows: detected foreground blob overlapping GT bounding box (TP), detected foreground object not overlapping a GT bounding box (FP), and a GT bounding box not overlapped by any detected object (FN). There are no definable true negatives.

<sup>4</sup>This must also capture the storage and post-event search costs

**Object Detection Rate:** Object detection rate (or true positive rate) measures the proportion of ground-truth objects correctly detected - see equation 1.

**False Alarm Rate:** False alarm rate (equation 2) determines the degree to which falsely detected objects (FP) dominate true objects (TP). In fact, tracking processes can robustly ignore most false objects but are especially vulnerable to false objects located near the position of the true object. We therefore redefine our *false alarm rate* as follows

$$f_a = \frac{N_{FP}^*}{N_{TP} + N_{FP}^*} \quad (8)$$

where  $N_{FP}^*$  is a count of falsely detected objects *near* to true objects. The degree of proximity  $\omega$  is of course an evaluation parameter which requires determining.

**Fragmentation Rate:** Fragmented targets present a subsequent tracker with the opportunity to incorrectly update a trajectory and subsequently fail to locate any observations. Our fragmentation rate  $\phi$  measures the number of observed blobs assigned as TP per GT object. False negative objects are not included in this metric. Thus

$$\phi = \frac{N_{TP}}{N - N_{FN}} \quad (9)$$

where  $N$  is number of GT objects and  $N_{FN}$  the number of GT objects without supporting detected blobs.

## 5. Optimising the Detection Algorithm

This section will determine the optimal parameter set for the motion detection algorithm described in Section 3 using the ROC methodology described in Section 4.2 applied to both the *Ticket Fraud* and *Evidence Gathering* scenarios. Using the metrics defined in Section 4.3, the optimised method will then be compared in Section 6 to two other algorithms reported in the literature - the Horprasert[21] algorithm and the Stauffer and Grimson method[22].

Figure 6 presents the ROC space populated with  $t_p, f_p$  pairs for all evaluated combinations of algorithm parameters for the proposed algorithm. These parameters were identified in Section 3 and listed in Table 2. Currently the search range of each parameter is evenly sampled seven times. Each of these points is evaluated using equation 6 to identify the optimal parameter set for the two scenarios. Optimal parameter values are reported in Table 2 for each scenario.

## 6. Comparative Evaluation

The optimised versions of the algorithm described in Section 3 are now compared with two well-known techniques reported in the literature: the Horprasert[21] algorithm and the Stauffer and Grimson method[22]. ROC analysis is also used to optimise the parameters of the Stauffer and Grimson for the two scenarios - see Figure 7. (Currently both optimised algorithms are compared to the published version of the Horprasert method. This is somewhat unfair as this too

Scenario	Evidence Gathering	Ticket Fraud
$\tau_B$	0.60	0.975
$\tau_S$	0.60	0.975
Updating	Adaptive	Adaptive
$\alpha_\mu^{\text{fast}}$	$1.0 \times 10^{-2}$	$2.0 \times 10^{-3}$
$\alpha_\Sigma^{\text{fast}}$	$1.0 \times 10^{-4}$	$4.0 \times 10^{-6}$
$\alpha_\mu^{\text{slow}}$	$5.0 \times 10^{-3}$	$1.0 \times 10^{-3}$
$\alpha_\Sigma^{\text{slow}}$	$2.5 \times 10^{-5}$	$1.0 \times 10^{-6}$
Thresholding	Constant	Linear
$\tau_A$	25	-
$\tau_{A_{\min}}$	-	50
$\gamma_A$	-	3

Table 2: Optimal Operating Points

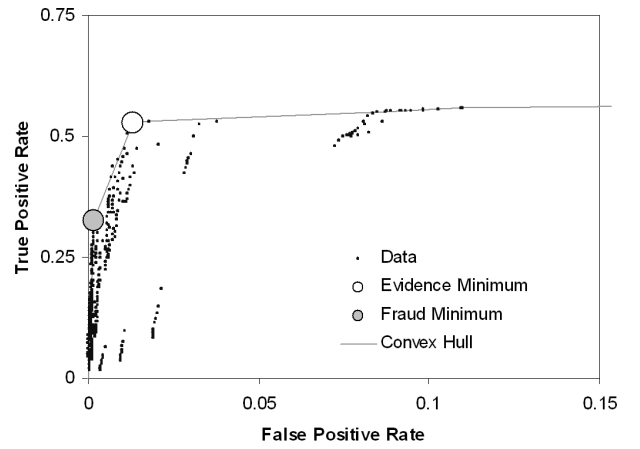


Figure 6: Determining Operating Points: Proposed Algorithm

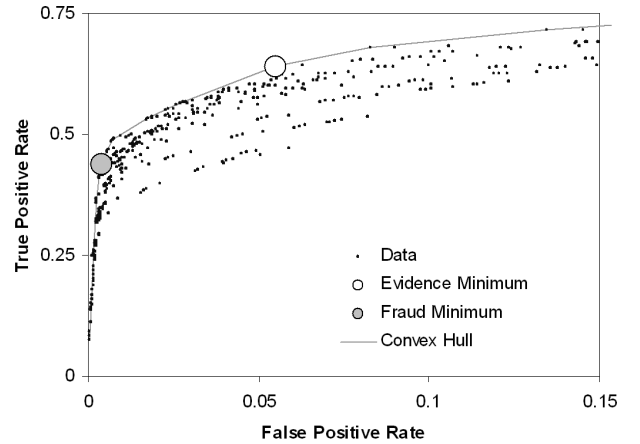


Figure 7: Determining Operating Points: Stauffer and Grimson

should be optimised for the specific scenarios.) Compared to our algorithm, the Stauffer and Grimson implementation performs a little better for both scenarios.

The classification costs of the algorithms are presented in Table 3. On the face of these results, Horprasert appears to outperform the Stauffer and Grimson and proposed methods in the evidence-gather scenario as the cost of misclassifying background pixels is relatively small. However in any real evidence gathering application using Horprasert would re-

Scenarios	Proposed	Stauffer	Horprasert
Ticket Fraud	0.0185	0.0177	0.8088
Evidence Gathering	2.483	2.320	0.8325

Table 3: Classification Costs

sult in recording everything! In the case of the Ticket Fraud scenario both the optimised Stauffer and Grimson and proposed methods perform significantly better than the Horprasert algorithm - not surprisingly as the latter has no specific adaption machinery.

To gain more insight, let us turn to the metrics displayed in Tables 4 and 5. The *Evidence Gathering* version of our method and the Stauffer and Grimson exhibit similar performance. About 99% of objects are located. From the localised FAR metric, each detected object has between fifteen and twenty surrounding noise blobs, and is often fragmented into one to two blobs. The *Ticket Fraud* does exhibit lower detection rates ( $\approx 94\%$ ) and an increased tendency for blob fragmentation. However, the amount of local confusion is very low - ideal for subsequent blob tracking purposes. Even more significant, however, is the *Specificity* metric which demonstrates the ability to successfully cope with the sudden, frequent and large lighting variations contained in the dataset.

Metric	Evidence Gathering		Ticket Fraud	
	Proposed	Stauffer	Proposed	Stauffer
SR	0.897	0.788	0.965	0.884
Spec	0.858	0.787	0.963	0.988

Table 4: Comparison of Pixel-based Metrics

Metric	Evidence Gathering		Ticket Fraud	
	Proposed	Stauffer	Proposed	Stauffer
DR	0.993	0.986	0.942	0.940
FAR	0.931	0.951	0.184	0.242
FR	2.286	1.581	3.623	2.931

Table 5: Comparison of Object-based Metrics

## 7. Conclusions

The primary purpose of this paper was to expose the surprisingly complex issues that arise when creating a well-designed evaluation methodology. We illustrated the process with a case study based on a novel contribution to motion detection through background modelling. The specific evaluation issues we explored were: good motivation of appropriate metrics; the distinction and relative merits of pixel-based versus object-based metrics; the need to define standardised application scenarios to provide context; the inevitable existence of evaluation parameters and the need to select appropriate values; and the role of ROC optimisation of algorithms. In the future we intend to address some of the current weakness: comparison with more methods

reported in the literature; a bigger range of datasets; and the need to evaluate motion segmentation by its impact on the performance of subsequent processing stages.

Though more computationally demanding, the Stauffer and Grimson algorithm has a greater capacity to represent multi-modal greylevel PDFs. However, sudden lighting conditions (and camera compensation) are not modelled well by a set of modes. Rather they are a dynamic process involving correlated changes in U,V. Although unimodal, the proposed method models the possible UV variations using a measure of the global colour content of the frame. Though explicitly optimised for specific application scenarios, the method performs well compared to two widely reported background modelling algorithms.

## References

- [1] P. Correia and F. Pereira. "Objective Evaluation of Relative Segmentation Quality". In *IEEE International Conference on Image Processing*, pages 308–311, Vancouver, Canada, September 10-13 2000.
- [2] L. Di Stefano, G. Neri, and E. Viarani. "Analysis of Pixel-Level Algorithms for Video Surveillance Applications". In *11th International Conference on Image Analysis and Processing, ICIAP2001*, pages 542–546, September 26-28 2001.
- [3] X. Gao, T.E. Boult, F. Coetzee, and V. Ramesh. "Error analysis of background Adaption". In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 503–510, 2000.
- [4] E.D. Gelasca, T. Ebrahimi, M.C.Q. Farias, M. Carli, and S.K. Mitra. "Towards Perceptually Driven Segmentation Evaluation Metrics". In *CVPR 2004 Workshop (Perceptual Organization in Computer Vision)*, page 52, June 2004.
- [5] Vladimir Y. Mariano, Junghye Min, Jin-Hyeong Park, Rangachar Kasturi, David Mihalcik, Huiping Li, David S. Doermann, and Thomas Drayer. "Performance Evaluation of Object Detection Algorithms". In *16th International Conference on Pattern Recognition (ICPR)*, volume 2, pages 965–969, 2002.
- [6] J. Nascimento and J. S. Marques. "New Performance Evaluation Metrics for Object Detection Algorithms". In *IEEE Workshop on Performance Analysis of Video Surveillance and Tracking (PETS'2004)*, May 2004.
- [7] P. Villegas, X. Marichal, and A. Salcedo. "Objective Evaluation of Segmentation Masks in Video Sequences". In *Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS'99)*, pages 85–88, May 1999.
- [8] P. Villegas and X. Marichal. "Perceptually Weighted Evaluation Criteria for Segmentation Masks in Video Sequences". *IEEE Transactions on Image Processing*, 13(8):1092–1103, August 2004.
- [9] J. Aguilera, H. Wildernauer, M. Kampel, M. Borg, D. Thirde, and J. Ferryman. Evaluation of Motion Segmentation Quality for Aircraft Activity Surveillance. In *Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, Beijing, October 15-16 2005.

- [10] A. Cavallaro, E.D. Gelasce, and T. Ebrahimi. Objective Evaluation of Segmentation Quality using Spatio-Temporal Context. In *IEEE International Conference on Image Processing*, page 301304, September 2002.
- [11] D. Hall, J. Nascimento, P. Ribeiro, E. Andrade, and P. Moreno. Comparison of target detection algorithms using adaptive background models. In *Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, Beijing, October 15-16 2005.
- [12] C.E. Erdem and B. Sankur. "Performance Evaluation Metrics for Object-Based Video Segmentation". In *X European Signal Processing Conference (EUSIPCO)*, September 4-8 2000.
- [13] J. Black, T.J. Ellis, and P. Rosin. "A Novel Method for Video Tracking Performance Evaluation". In *IEEE Workshop on Performance Analysis of Video Surveillance and Tracking (PETS'2003)*, pages 125–132, October 2003.
- [14] C.E. Erdem, A.M. Tekalp, and B. Sankur. "Metrics for performance evaluation of video object segmentation and tracking without ground-truth". In *IEEE International Conference on Image Processing (ICIP)*, October 7-10 2004.
- [15] B. Georis, F. Bremond, M. Thonnat, and B. Macq. "Use of an Evaluation and Diagnosis Method to Improve Tracking Performances". In *IASTED 3rd International Conference on Visualization, Imaging and Image Processing*, September 8-10th 2003.
- [16] C. Jaynes, S. Webb, M. Steele, and Q. Xiong. "An Open Development Environment for Evaluation of Video Surveillance Systems". In *IEEE Workshop on Performance Analysis of Video Surveillance and Tracking (PETS'2002)*, June 2002.
- [17] T. Schlogl, C. Beleznai, M. Winter, and H. Bischof. "Performance evaluation metrics for motion detection and tracking". In *17th International Conference on Pattern Recognition (ICPR)*, volume 4, pages 519–522, August 23-26 2004.
- [18] H. Wu and Q. Zheng. "Self-evaluation for video tracking systems". In *Proceedings of the 24th Army Science Conference*, November 2004.
- [19] Thor List, Jos Bins, Jose Vazquez, and Robert B. Fisher. "Performance Evaluating the Evaluator". In *IEEE Joint Workshop on Visual Surveillance and Performance Analysis of Video Surveillance and Tracking (VS-PETS 2005)*, 15-16th October 2005.
- [20] R. Fisher. Caviar - context aware vision using image-based active recognition. In <http://homepages.inf.ed.ac.uk/rbf/CAVIAR>.
- [21] T. Horprasert, D. Harwood and L.S. Davies. "A Robust Background Subtraction and Shadow Detection". In *Asian Conference on Computer Vision (ACCV2000)*, pages 8–11, January 8-11 2000.
- [22] C. Stauffer and W.E.L. Grimson. "Adaptive background mixture models for real-time tracking.". In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'2000)*, pages 246–252, Fort Collins, Colorado, June 23-25 1999.
- [23] S.J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler. "Tracking Groups of People". *Computer Vision and Image Understanding*, 80(1):4256, October 2000.
- [24] D. Greenhill, J.R. Renno, J. Orwell, and G.A. Jones. "Learning the Semantic Landscape: Embedding scene knowledge in object tracking". *Real Time Imaging*, 11:186–203, 2005.