# Performance Evaluation of Surveillance Systems Under Varying Conditions

Lisa M. Brown, Andrew W. Senior, Ying-li Tian, Jonathan Connell, Arun Hampapur,
Chiao-Fe Shu, Hans Merkl, Max Lu
IBM T.J. Watson Research Center, lisabr@us.ibm.com

## Abstract

*Effectively evaluating the performance of moving object detection and tracking algorithms is an important step towards attaining robust digital video surveillance systems with sufficient accuracy for practical applications. As systems become more complex and achieve greater robustness, the ability to quantitatively assess performance is needed in order to continuously improve performance. In this paper, we refine the methods used to estimate performance and use these methods to measure the performance of our system under several different conditions including: indoor/outdoor, different weather conditions (precipitation, wind, and brightness), different cameras/viewpoints, and as a standard benchmark, the PETS 2001 datasets. To test the extensibility/validity of our results, we have also evaluated our system on four longer data sets (20-30 min each) from four different cameras. We evaluate the performance of the background subtraction alone and with a simple tracking system using two different sets of metrics. Visualization of the performance results has proven critical for understanding the weaknesses of the system.*

### 1. Introduction

In practice, digital video surveillance needs to operate around the clock, as the weather varies, the seasons change, and the daily events unfold. Performance evaluation of automatic surveillance systems is still typically limited to short sequences. Unfortunately, these sequences and their annotation are often available only to the researchers who created them. There is a need to create benchmark datasets available to all researchers and to agree on standardized performance metrics for their evaluation. Furthermore, we need to understand the extensibility of these results to the long-term operation of systems for real-world applications. In an effort to begin to understand the issues involved in running real-time real-world around-the-clock digital video analysis, we have developed and evaluated a test bed of sequences, across a range of conditions. The conditions we have studied include: indoor/outdoor, varying weather conditions, and different cameras/viewpoints. We have also evaluated our system on the standard datasets provided by the PETS 2001 workshop.

In this paper, we suggest several metrics for system performance evaluation, and test our system on these metrics for several different types of sequences. When possible, we provide the sequences and their ground truth annotation online for general accessibility and further testing. [http://www.research.ibm.com/peoplevision/ performanceevaluation.html]

Performance evaluation systems have been developed to analyze the two primary levels of processing: background subtraction and tracking. We believe these are both important elements of any digital video surveillance system. However, since the ultimate performance of such a system, relies on correct tracking and subsequent triggering of specific alarms and possibly archiving the correct video clips, it is important to distinguish raw background subtraction detection accuracy from the eventual high level tracking and triggering of an event alarm. Background subtraction may influence the effectiveness of the tracking, but the final tracking will reflect the system's capabilities.

In the next section of the paper, we discuss the state-of-the-art in performance evaluation including the absolute performance of systems – i.e., what are the current performance values reported by researchers on their systems. In Section 3, we describe the annotation process used to generate ground truth. In Section 4, we describe our background subtraction evaluation system. In Section 5, we explain our track evaluation system. In Section 6, we describe the datasets used for evaluation. In Section 7 we show the results on each of the datasets. Finally we give our conclusions in Section 8.

### 2. Background

Some of the earliest efforts in performance evaluation of moving object detection and tracking began at the

PETS (Performance Evaluation of Tracking and Surveillance) workshops. In 2000 and 2001, the workshop provided general outdoor surveillance benchmark datasets for the participants to evaluate their systems. [Senior 01] suggested several metrics for evaluating performance of tracking including: # track false positives, # track false negatives, average position error, average area error, average detection lag, and average track incompleteness. However, the results of these metrics are highly dependent on the input sequence and practical systems need to perform well on a wide range of input data.

[Toyama 99] was the first to analyze the performance of nine background subtraction methods using the number of pixels erroneously classified as foreground (false positives) or not detected (false negatives). The results were based on the manual and somewhat arduous annotation of seven short (several minutes) sequences using the outline of each foreground object at 4Hz and a resolution of 160x120.

The sequences were chosen to exemplify each of 7 challenging situations for background subtraction methods. In particular, the situations were: moved object (static objects are moved), time of day (gradual lighting change), light switch (sudden light change), waving trees (uninteresting motion), camouflage (foreground similar to background), bootstrap (constant motion, no time to learn background), and foreground aperture (uniformly colored object moves). Their results show the comparative performance of the nine algorithms and the advantages of background subtraction methods which determine foreground objects based on spatially varying criteria (i.e., not just based on pixel level models.) For further comparison, their sequences need to be made accessible to other researchers.

More recently (at PETS 03), two new methods have been proposed to evaluate background subtraction and in the second case, tracking performance. [Chalidabhongse03] proposed a background subtraction evaluation system in which the false alarm rate (FAR) is fixed, typically in the range of .1-.01 percent of pixels/frame, and uniform random contrast differences are generated to determine the just noticeable difference (JND) for background subtraction. They compared 4 methods and found their own codebook approach which uses a nonparametric quantization/clustering model to be superior. Their analysis compares the raw performance differences of pixel-based approaches. This is a useful, repeatable (although of course, it depends on the sequences) metric but limited to measuring raw (i.e. pixel level) models. The degree to which the role of the base performance of background subtraction pixel-level model effects the ultimate results of tracking 24/7 in the real world are still unclear.

[Black 03] propose a methodology to minimize the painstaking manual annotation necessary to provide accurate ground truth information. They suggest generating a range of tracking situations based on a small set of manually annotated tracks by using different combinations of tracks on different background scenes. Although this clearly eliminates significant labor, it is not clear this will simulate many of the issues present in actual video such as wind and shadows or effects due to inaccurate background subtraction healing (e.g. “ghosts” left behind when stationary objects begin to move.) Most of these effects are complexly related to the tracks themselves and will not occur with simulation but may severely affect performance. Furthermore there is a need to discover these real-world issues.

They also report the performance of their system on the PETS 2001 data for one sequence (dataset 2, camera 2). For the full resolution (768x576) color video, they report a false alarm rate of .01 and a track detection rate of 98%. In terms of our metrics (described in Section 4, this is equivalent to FP=.02, FN=.02, and track fragmentation=1.2. This is one of the few quantitative and comparable results reported for a publicly available sequence albeit for a single sequence and no timing information.

System to ground truth alignment was based on minimizing the distance metric (match each system track to its best GT track) weighted inversely by the length of the temporal overlap [Senior01]. We will discuss this metric and its limitations in Section 5. [Pingali96] describe an approach to measure tracking accuracy based on trajectories and trajectory events such as crossings. Matching system to ground truth tracks is based on comparing trajectories and trajectory events. Their primary contribution allows performance evaluation of specific application goals such as counting.

As far as we know, the work of [Black04], [Pingali96] and [Senior01] are the only papers which quantitatively evaluate the performance of full tracking algorithms, i.e., tracking of multiple objects through occlusions. Other researchers have studied tracking systems only with regard to segmentation accuracy, i.e., without evaluating occlusions, merging and splitting. [Tissainayagam02] evaluated the performance of contour trackers but they only consider performance of the trackers in terms of the accuracy of the contour and assume a single object without occlusion. [Erdem04] measure tracking performance based on segmentation accuracy using spatial differences of color and motion along the boundary. [Dahlkamp04] visually compare two vehicle tracking methods by analyzing their respective failure modes and comparing the time intervals in which each vehicle is successfully tracked.

In this paper, we describe a method to evaluate tracking performance for real-world situations in which multiple objects traverse the scene, there is significant background clutter, and objects are occluded by the scene and each other.

## 3. Ground Truth Acquisition

Ground Truth (GT) Tracks were obtained manually using an annotation tool. Annotation is performed every 30 frames and at start/end of each track. The user draws the appropriate bounding box around each foreground object which is associated with a track. If the object is temporarily predominantly occluded, the user marks it as such. The system tracks are obtained by an automatic script. More information about our system, the Smart Surveillance System can be found in [Hampapur03].

## 4. Background Subtraction Evaluation

The background subtraction evaluation compares every ground truth frame against the results of a specific background subtraction algorithm. Each comparison determines if there is a false negative (FN): no system foreground object centroid inside the ground truth bounding box or false positive (FP): system foreground object does not intersect with any ground truth bounding box. If a foreground moving object becomes stationary, we do not measure performance for this region i.e., whether the system continues to detect this object for longer or no longer detects it, we do not consider it to be either a FP or FN because of the ambiguity of the situation.

The evaluation determines the number of true positives (TP) over all ground truth frames. The final FP measure represents the average number of false positives per ground truth frame. The final FN measure is the percentage of TP which are missed by the system. For both false positives and false negatives, we also measure the average area (in pixels) of their respective instances. These values are referred to as FPSize and FNSize.

We compare two background subtraction algorithms. The first uses a variation of the adaptive mixture of Gaussians model (MOG) [Stauffer 99]. We use three Gaussians per pixel and a threshold of .3. The multi-adaptive model learning rate is .01 and the weight update learning rate is .005. The second method, we call Salience-Based (SAL) and is described in [Connell04]. This method combines evidence from differences in color, texture and motion. The method also has several built-in mechanisms to handle changing ambient conditions and scene composition. First, it continually updates its overall RGB channel noise parameters to compensate for changing light levels. Second, it estimates and corrects for automatic gain control and white balance shifts induced by the camera. Finally, it maintains a map of high activity regions and slowly updates its background model only in areas deemed as relatively quiescent.

## 5. Tracking Evaluation

In addition to measuring the performance of the background subtraction, we also measure the performance of the full system (background subtraction followed by tracking.) For this evaluation, we need to determine which system tracks correspond to which ground truth tracks.

In [Senior 01], the evaluation matched system tracks to ground truth tracks. The correspondence was many-to-one, i.e., several system tracks could be matched to one ground truth track but not vice versa. A match was based on proximity and the overlap duration:

$$\frac{\sum_{i}^{MatchDuration} Dist(p_i^{GT}, p_i^{Sys})}{MatchDuration * MatchDuration}$$

where $p_i^{GT}$ is the centroid of the ground truth track at the $i^{\text{th}}$ ground truth frame, $p_i^{Sys}$ is the centroid of the system track, and *Dist* is the Euclidean distance. MatchDuration is the number of frames in the overlap. If the match score is below a threshold, the two tracks are matched.

This metric is useful for simple scenes and tracking scenarios but has several limitations. This can be best explained in terms of the four types of tracking errors: spatial fragmentation, temporal fragmentation, spatial merging and temporal merging. Fig 1 shows examples of track fragmentation error. This can be due to either spatial error (e.g. a single person results in an upper and lower body track) or temporal (e.g. a small object is only intermittently observed). In the latter case, the horizontal axis represents time. Fig 2 shows examples of track merge error. Temporal merging is often due to the track of one object exiting just as the track of another object enters the scene. Spatial merging is often the result of the tracks of two objects merging when they appear close together.

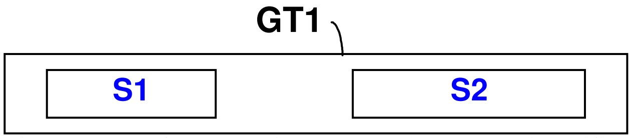

Fig 1. Track fragmentation error. The system identifies multiple tracks for a single real track.

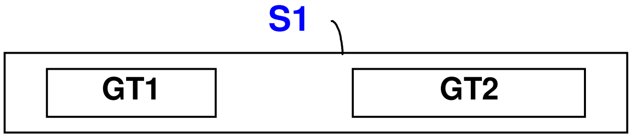

Fig 2. Track merge error. The system identifies a single track for multiple real tracks.

The problem with matching tracks using the previously described match score based on proximity is twofold. Proximity between the centroids of the system and ground truth tracks is often insufficient to correctly match tracks when multiple tracks are nearby; size and shape information should also be considered. Second, since a many-to-one match is performed (many ground truth tracks to each system track) only track fragmentation errors are addressed. We have found when evaluating a wide range of sequences, that this metric is inadequate.

We propose a new two-pass matching scheme to address these limitations as seen in Figure 3. In phase 1, each system (S) track is allowed to match to many ground-truth (GT) tracks. A GT track is matched to the system track if there is both temporal overlap and spatial overlap. Temporal overlap is with respect to the duration of the system track. Spatial overlap is based on the centroid of the system lying inside the bounding box of the ground truth track. If multiple GT tracks are matched to a particular system track, then the cumulative temporal/spatial overlap is computed, i.e, percent of frames which overlap both spatially and temporally. This is used to find track false positives i.e. system tracks with insufficient matches. We threshold the cumulative overlap to identify system tracks with "insufficient" matches (track false positives). By measuring temporal and spatial overlap, we address the problems of temporal and spatial merging respectively. After this matching phase is completed, we can find instances of track merging – system tracks which are explained by multiple ground truth tracks.

In phase 2, each GT track is matched to many system tracks. This is used to determine track false negatives, i.e., ground truth tracks with insufficient matches. In this case, temporal and spatial overlap is used to identify instances of temporal and spatial fragmentation respectively. By determining spatial overlap based on the system track centroid location inside the GT track bounding box (or vice versa for phase 1), we have a more precise estimate of track coincidence than proximity. We enlarge the bounding box by 20% (E1=E2=.2) to account for small errors in segmentation. After this matching phase is completed, we can find instances of track fragmentation – ground truth tracks which are explained by multiple system tracks.

After track matching is performed, it is possible to measure the number of "track" false negatives (TFN) and track false positives (TFP). These should be tracks which have either been missed by the system or incorrectly found by it. We have set the temporal overlap thresholds for TFP to T1=.5 and for TFN to T2=.01. Visualizing these tracks has been very useful in understanding the causes of these problems and the ultimate system performance.

In addition to measuring the number of track false positives (TFP) and track false negatives (TFN), we also measure the average size and duration of TFPs and TFNs. The fragmentation error is defined as the number of system tracks per ground truth track (phase 1). The merge error is defined as the number of GT tracks per system track (phase 2).

## 6. Data Sets

**PETS 01** – 4 sequences from the Performance Evaluation in Tracking for Surveillance (PETS) Workshop 2001, 2 different cameras of an outdoor campus scene, high quality (from digital camera), with resolution (358,288) 30fps, stored as avis with no compression. The data from PETS01 was originally of higher resolution and stored as JPEG images.

**Hawthorne Outdoor** – 10 sequences from an IBM building entrance and parking lot, from 4 different Sensormatic NTSC cameras, many different viewpoints, range of NY weather conditions, 320x240 resolution, 30fps, MPEG1 compressed, 2-5minutes each.

**Longer Sequences –** 4 longer sequences from 4 different IBM Sensormatic NTSC cameras, 320x240 resolution, 30fps, MPEG1 compressed, 20-30 minutes each. Significant lighting changes and windy conditions including camera instability

**Indoor –** 11 sequences, 5 different NTSC cameras, 320x240 resolution, 30fps, MPEG1 compressed, less than 3 minutes each. Three sequences are taken simultaneously by 3 cameras in our laboratory as two or three people walk by and around each other. Two other sequences are taken from two different cameras in our lobby and by our elevators, of two people, each walking along a corridor, following one another, and then walking past each other.

## 7. Results

We first report our results on the PETS01 datasets. Tables 1 and 2 show the results of the background subtraction and tracking evaluation for the two different background subtraction methods. We compare the results for varying resolution (ds1 = full resolution, ds2= half resolution), varying the minimum connected component size threshold (30 or 100 pixels for full resolution), and for grey-scale (8-bit) vs. color (RGB – 24bit). In each case (resolution, component size threshold and grey vs. color) there is a clear trade-off between improved detection (lower FN) and over-sensitivity (increased FP). This relationship is depicted in the plots shown in Figure 4. These plots show the relationship between the false negatives and false positives for each background subtraction method. Each line segment represents the two values obtained at the two resolutions (ds1/ds2). In addition to showing the trade-off between FN/FP, the plot for the salience-based approach indicates little change with grey to color or from low (ds2) to high (ds1) resolution This is not true for the MOG method.

Table 2 shows the average number of FP per frame based on the background subtraction evaluator and the number of track false positives and their average duration based on the track evaluator. Although the number of false positives is high, the average duration is typically short (<100 frames or 3 seconds). The percentage of false negatives with respect to the total number of true positives and their average size based on the background subtraction evaluator is also shown. In addition, the number of track FN and their average size per frame in square pixels are also given. Track FN are typically less than 100 square pixels.

The best results are obtained using the SAL method, at half resolution (DS2), with minimum connected component size of 30 and color pixels. At these settings, there were 6 TFP and 8 TFN. The evaluator automatically creates a video of these tracks for visualization of the results. Figure 4 shows an example frame from the TFPs and one of the TFN. The other FNs are very similar – along the same distant partially occluded road. The TFP are due to (1) parked cars beginning to move leaving a "hole" (2) moving tree and moving object resulting in "extra" object and (3) shadows.

Table 3 shows the results of the background subtraction and tracking evaluation on the 10 Hawthorne outdoor sequences using the MOG background subtraction method, CCMin=100, and grey-scale pixels. MOG performed modestly better than SAL overall. Figure 5 shows several examples of track FPs and FNs.

Table 4 shows the results of the evaluation on the four longer outdoor sequences. It can be seen with more data, our results are not yet sufficient to generalize. For these longer sequences the SAL method was significantly more robust to the strong lighting changes which caused innumerable FP for MOG method.

Table 5 shows the results of the evaluation on the 11 indoor sequences. The moving objects in the indoor data were substantially larger than outdoors and not subject to the lighting changes, weather and camera motion due to wind. Hence the indoor data had no TFN and only one TFP due to shadows. For indoor data, the performance was most influenced by the accuracy of tracking through occlusion. For our simple appearance-based tracker there were significant amounts of fragmentation and merging. Some of the merging is due to actually merging of GT tracks. In this case, a system track will correctly match to multiple GT tracks. This type of merging should not be reported as merge "errors." Figures 7 and 8 show examples of track fragmentation and merge errors. Figure 7 shows an example of spatial merging and temporal merging. Figure 8 shows an example of temporal fragmentation and temporal merging due to track crossing (the system follows one track then loses this track which continues and incorrectly follows a different track.) For the appearance based tracker used in these examples – spatial fragmentation did not occur.

The time required by the system (background subtraction and tracking) to process each frame for a given video sequence (from the 10 outdoor Hawthorne videos) is shown in Figure 6. This is based on MOG background subtraction followed by appearance based tracking. The graph is a histogram showing the relative frequency of frame times in microseconds on a 2.4 GHz machine. This plot shows that most frames take < 12ms to process and very few take more than 20ms (~50fps). The left hand peak (7ms/frame) corresponds to frames in which no foreground is detected. The right hand (and broader) peak corresponds to frames in which tracking must be carried out in addition to background subtraction.

## 8. Conclusions

In this paper we have presented a new method for evaluating the performance of background subtraction and tracking including a track evaluation based on matching ground truth tracks to system tracks in a two-way matching system. We have shown the quantitative results of this evaluation on the PETS benchmark data, over 100 minutes of outdoor data with a wide range of camera viewpoints, weather conditions, lighting changes and camera instability. We have also shown results on indoor data.

We have made some of the data and annotation available publicly (when possible) in order to enable the community to work together to understand the relative merits of different algorithms. Consequently many of our results can be openly compared to results with other algorithms. We have illustrated the trade-off between FN and FP detection based on varying background subtraction method, resolution, color vs. grey-scale, and the minimum connected component size. But, we have also shown, via the use of longer sequences, that insufficient data is yet available to determine the performance of systems for around-the-clock operation.

## 9. References

**[Black03]** Black, J. et al., "A Novel Method for Video Tracking Performance Evaluation," ," Joint IEEE Int'l Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS), Nice France, October 11-12, 2003, p125-132.

**[Chalidabhongse03]** Chalidabhongse, T.H., et al., "A Perturbation Method for Evaluating Background

Subtraction Algorithms," Joint IEEE Int'l Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS), Nice France, October 11-12, 2003, p110-117.

**[Connell04]** Connell, J., "Detection and Tracking in the IBM PeopleVision System," IEEE ICME, June 2004.

[Erdem 04] Erdem, C.E. et al., "Performance Measures for Video Object Segmentation and Tracking," IEEE Trans. On Image Processing, Vol 13, No. 7, July 2004.

**[Dahlkamp04]** Dahlkamp, H. et al., "Differential Analysis of Two Model-Based Vehicle Tracking Approaches," DAGM 2004, LNCS 3175, pp71-78, 2004.

**[Erdem04]** Erdem.C. et al., "Performance Measures for Video Object Segmentation and Tracking," IEEE Trans. On Image Processing, Vol 13, No. 7, July 2004.

**[Hampapur03]** Hampapur, A. et al., "Smart Surveillance: Applications, Technologies and Implications," IEEE Pacific-Rim Conference On Multimedia, Singapore, Dec. 2003.

**[Pingali96]** Pingali, S. and Segen, J., "Performance Evaluation of People Tracking Systems," IEEE Workshop on Applications of Computer Vision, p33-38, 1996.

**[Senior01]** Senior, A. et al., "Appearance Models for Occlusion Handling," IEEE Int. Workshop on Performance Evaluation of Tracking and Surveillance, Kauai, HI, December 9, 2001.

**[Stauffer99]** Stauffer, C. and Grimson, W.E.L., "Adaptive Background Mixture Models for Real-time Tracking," Int'l Conf. Computer Vision and Pattern Recognition, Vol. 2, pp246-252, 1999.

**[Tissainayagam02]** Tissainayagam, P., and Suter, D., "Performance Measures for Assessing Contour Trackers," IEEE Int. Journal of Image and Graphics, Vol 2, p343-359, April 2002.

**[Toyama99]** Toyama, K. et al., "Wallflower: Principles and Practice of Background Maintenance," Seventh Int'l Conf on Computer Vision, pp255-261, 1999.

1. **System-Track-Matching** – for every system track find all "GT-matches"

    "GT-match" = Temporal-Overlap AND Spatial-Overlap

    Temporal-Overlap = overlap/(system duration)

    Spatial-Overlap = GT centroid inside E1% enlarged system bounding box

    If cumulative temporal/spatial overlap < T1, then system track has insufficient matches and is labeled a FP.

    If multiple GT-matches, then this system track has merge error = # matched GT tracks

2. **GT-Track-Matching** – for every GT track find all "system-matches"

    "System-match" = Temporal-Overlap AND Spatial-Overlap

    Temporal-Overlap = overlap/(GT duration)

    Spatial-Overlap = system centroid inside E2% enlarged GT bounding box

    If cumulative temporal/spatial overlap < T2, then GT track has insufficient matches and is labelhbghed a FN.

    If multiple system-matches, then this GT track has fragmentation error = # matched Sys tracks

**Figure 3.** Two-pass many-to-many system to ground truth (GT) track matching criteria

| | | **MOG 30** | | **MOG 100** | | **SAL 30** | | **SAL 100** | |
|---|---|---|---|---|---|---|---|---|---|
| | | **FP** | **FN** | **FP** | **FN** | **FP** | **FN** | **FP** | **FN** |
| **COLOR** | **DS1** | .48 | 7.8 | .11 | 20.2 | .11 | 19.9 | .07 | 27.1 |
| | **DS2** | .22 | 13.3 | .07 | 23.6 | **.07** | **20.7** | .05 | 27.3 |
| **GREY** | **DS1** | .11 | 11.7 | .05 | 25.8 | .14 | 20.7 | .08 | 28.5 |
| | **DS2** | .09 | 19.7 | .06 | 30.2 | .09 | 20.7 | .04 | 28.6 |

**Table 1**. Performance results on PETS01 data – using different background subtraction methods (SAL/MOG), different resolution(DS1/DS2), and minimum connected component size (30 or 100 pixels) for Color/Grey-Scale data.

| **File** | **Frames** | **True Positives** | **False Positives** | **False Negatives** | **Track TP** | **Track FP** | **TFP duration** | **Track FN** | **TFN area** |
|---|---|---|---|---|---|---|---|---|---|
| MOG | 1120 | 1120 | .11 | 11.7 | 90 | 15 | 70 | 0 | - |
| SAL | 2267 | 1120 | .07 | 20.7 | 90 | 6 | 59 | 8 | 90 |

**Table 2.** Performance results including performance of tracking on PETS01 data – using two best results (SAL-color-ds2 and MOG-grey-ds1.

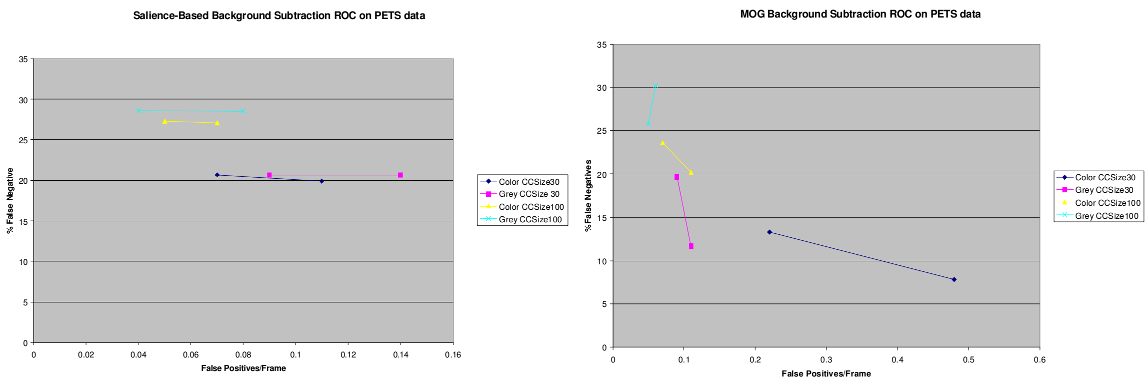

**Figure 4.** ROC plots of the performance of two background subtractions plots. Each line segment represents the results at two resolutions. Performance varies from upper left (high FN, low FP using grey-values, low resolution, and large size threshold to bottom right (low FN, high FP) for color, high resolution and small size threshold.

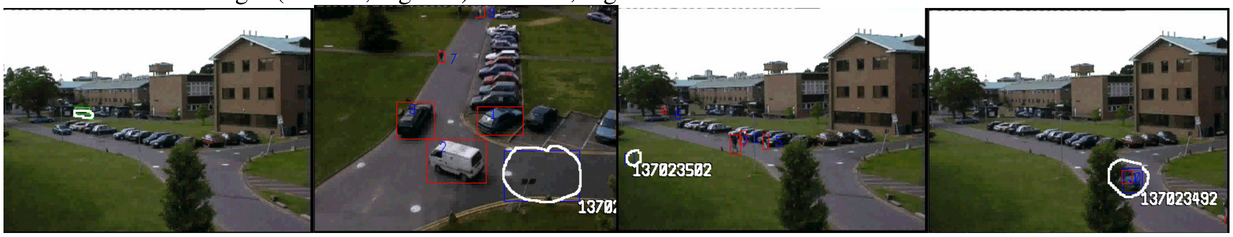

**Figure 5.** Top left, track FN example – car behind trees on upper left. Top right, track FP due to stationary truck moving and leaving a "ghost." Bottom left, track FP due to moving tree near moving object. Bottom right, track FP due to shadow on grass (number on image where FP occurred.)

| File | Frames | True Positives | False Positives | False Negatives | Track TP | Track FP | TFP duration | Track FN | TFN area |
|---|---|---|---|---|---|---|---|---|---|
| MOG | 2267 | 2964 | .03 | 17.7 | 90 | 6 | 79 | 12 | 110 |
| SAL | 2267 | 2964 | .03 | 21.2 | 90 | 11 | 65 | 14 | 137 |

**Table 3.** Performance Results on 10 Hawthorne Outdoor Videos (CCSize100,ds2,grey)

| File | Frames | True Positives | False Positives | False Negatives |
|---|---|---|---|---|
| 1 | 1704 | 836 | .022 | 8.6 |
| 2 | 1595 | 497 | .015 | 7.7 |
| 3 | 1320 | 1492 | .072 | 25.1 |
| 4 | 1320 | 543 | .054 | 22.3 |

**Table 4.** Performance results on 4 longer Hawthorne outdoor sequences – using SAL background subtraction.

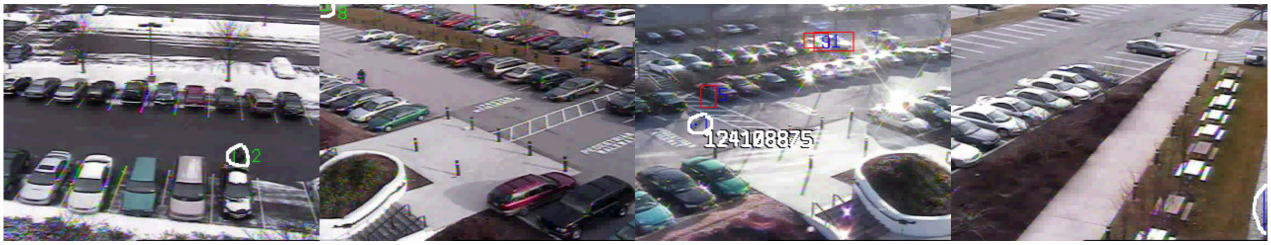

**Figure 6.** Two pictures on left are example frames from track FNs - the first is not detected because of insufficient contrast, the second because it lies on the border of the video image (upper left). Two pictures on right are examples from track FPs – the first is the result of significant shadows, the second from reflections off glass of building (bottom right of image.)

| Indoor Data | GT Frames | TP | FP | FN | Tracks | TFP | TFN | #sys/GT | #GT/sys |
|---|---|---|---|---|---|---|---|---|---|
| Total /Ave | 254 | 218 | .01 | 3.7 | 48 | 2 | 0 | 1.5 | 1.25 |

**Table 5.** Performance results on indoor data including track fragmentation and merge errors.

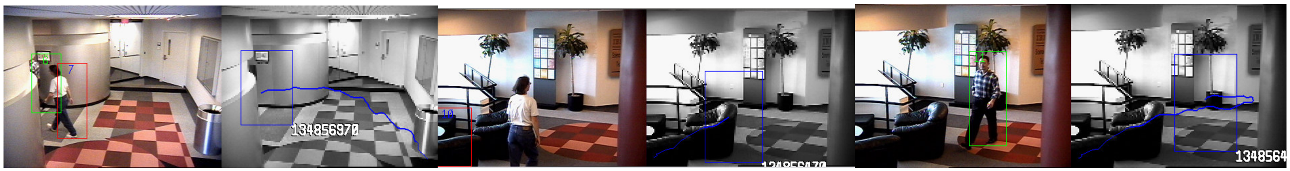

**Figure 7.** Left two images: spatial merge – system combines two people into one track, Middle two images: 1st person walks across, Last two images: 2nd person then walks, causing temporal merge – system combines tracks of both people, one after the other.

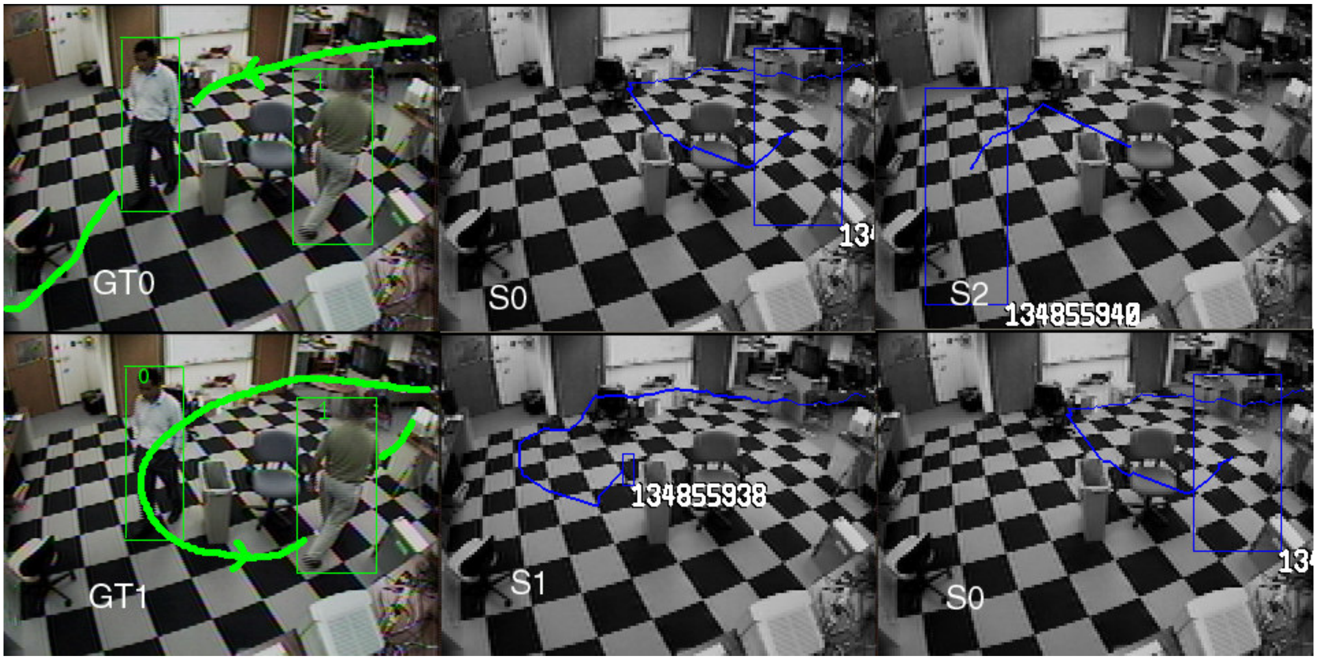

**Figure 8.** Top Row – first person walks halfway across room, stops and then continues. System is confused when the person stops and creates another track when he restarts resulting in temporal fragmentation (Tracks S0 and S2). Second Row – second person walks around the first person. System initially tracks this person as S1 but then incorrectly connects his final exit to track S0.

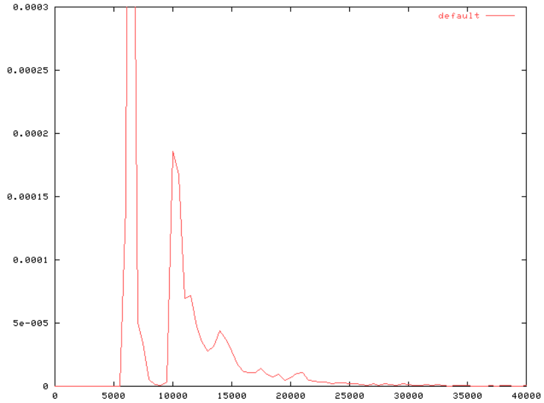

**Figure 9.** Histogram showing the relative frequency of frames execution times (in microseconds on a 2.4GHz machine). The left hand peak (7ms/frame) corresponds to frames in which no foreground is detected. The right hand (and broader) peak corresponds to frames in which tracking must be carried out in addition to background subtraction.