

# CRIM Notebook Paper - TRECVID 2010

## Surveillance Event Detection

S. Foucher, *Member, IEEE*, M. Lalonde, *Member, IEEE*, L. Gagnon, *Member, IEEE*

Centre de recherche informatique de Montréal (CRIM)  
{Samuel.Foucher, Marc.Lalonde, Langis.Gagnon}@crim.ca

### Abstract

*Approach we have tested in each of your submitted runs.* Only one run was provided for our first year in this competition. Our system is an adaptation of an outdoor video-surveillance system composed of a real-time blob tracker and an offline scene understanding module that accumulates statistics about observed objects over a long period of time. In particular, statistics about object height and velocity are accumulated over time using a non parametric approach. For the “Object Put” event, we followed a dual foreground segmentation approach where the output difference between a short term and a long term model is used for triggering potential alerts. For Pointing, we applied the learning of compound spatio-temporal features based on a data mining method.

*Relative contribution of each component of our approach.* From the results, the tracking system, originally designed for outdoors scenes, appears to be the weak component in our system. We need to improve the background/foreground segmentation in order to produce less fragmented objects. Also, we don’t have a pedestrian or upper-body detector available this year (it is planned for next year however) so we are tracking many foreground objects that are not pertinent.

*What we learned about runs/approaches and the research question(s) that motivated them.* For the *PersonRuns* task, we adopted a simple non parametric approach where we are looking for velocity outliers on trajectories provided by our tracker. Results were not up to our expectations mainly because our object tracker is not performing well enough. For the *ObjectPut* task, we were looking for a fast and low level approach that could detect static objects appearing in the image foreground. It produced results beyond our expectations. The approach, however, is not able to separate static persons versus real static objects leading to a high number of false alarms. For “*Pointing*”, we implemented a recent method based on a data mining of spatio-temporal grouping of local corners. This method demonstrated promising results on action recognition datasets but has never been applied before in a video surveillance application.

### Introduction

This is the first year of participation for CRIM so we only provided results on three events. Our previous work on video surveillance [1] was directed mainly on outdoor surveillance systems which involve very different constraints. For the *ObjectPut* and Pointing events, we developed separate algorithms that are independent of our tracking system.

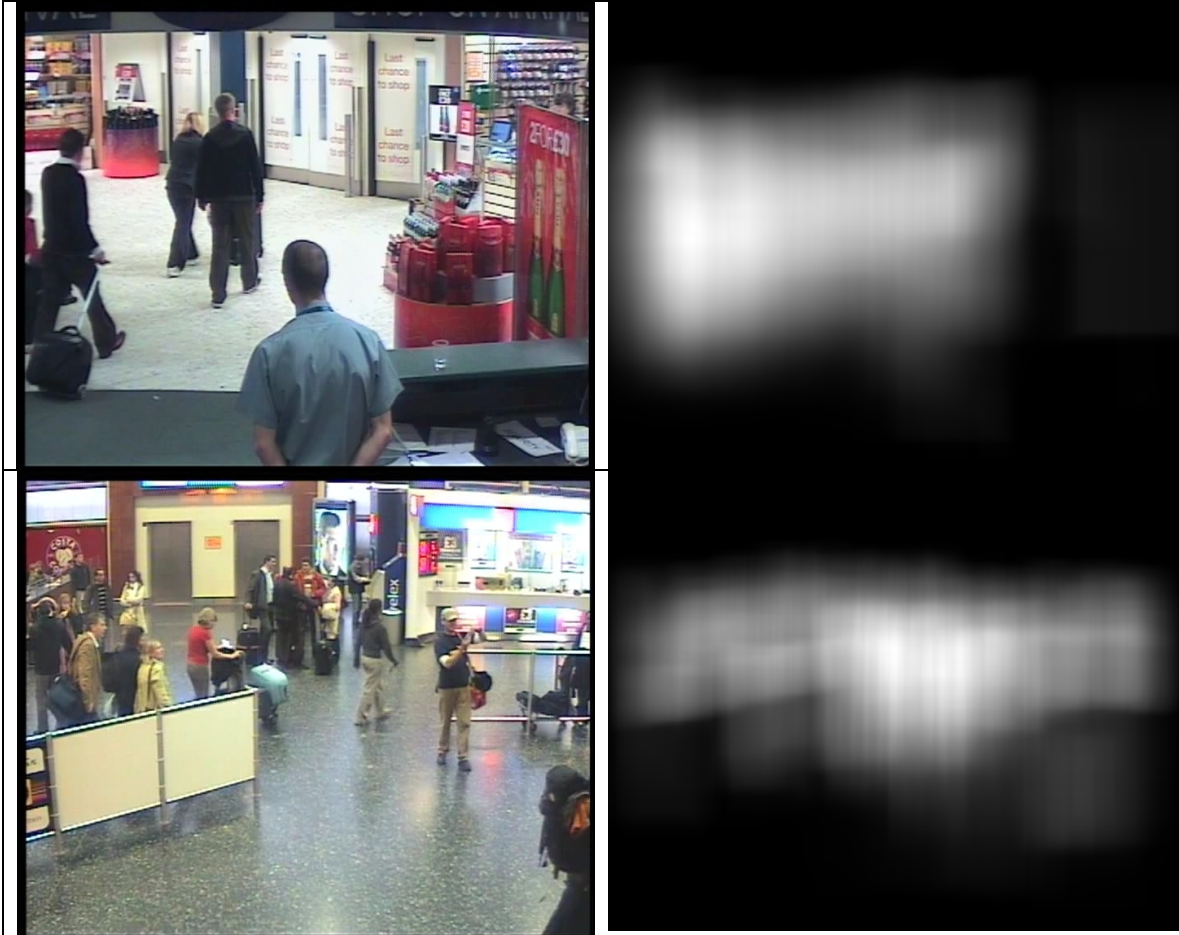
All the computations were performed on the “Mammoth” supercomputer located at the Center for Scientific Computing at the Université de Sherbrooke.

## I – Scene Modeling and Understanding

### Pedestrian occurrence

The goal here is to produce processing masks for the various tasks as well as build simple camera geometry models in order to reduce false alarms. Pedestrian detections were performed on the entire development set

(100 hours) using the Dalal and Triggs detector [2]. For each position within the scene, we estimated also the average pedestrian height. The height measurements are also exploited for the geometric modeling of each camera view. A Pedestrian probability map is also computed as shown in Figure 1. Those probability maps, once thresholded, will define the processing masks for the “PersonRuns” event detection.



**Figure 1.** top row: a frame from Camera 1 and the corresponding pedestrian probability map (white= high number of detection occurrences); bottom row: same for Camera 3.

### Camera Geometry

A similar approach was used by the SFU team at TRECVID 2009 [4]. Assuming a simple projective geometry, a camera parallel to the ground plane and objects only on the ground plane, we can exploit the following relationship between the real world object height  $h$  and the observed image height  $\Delta y$  [3]

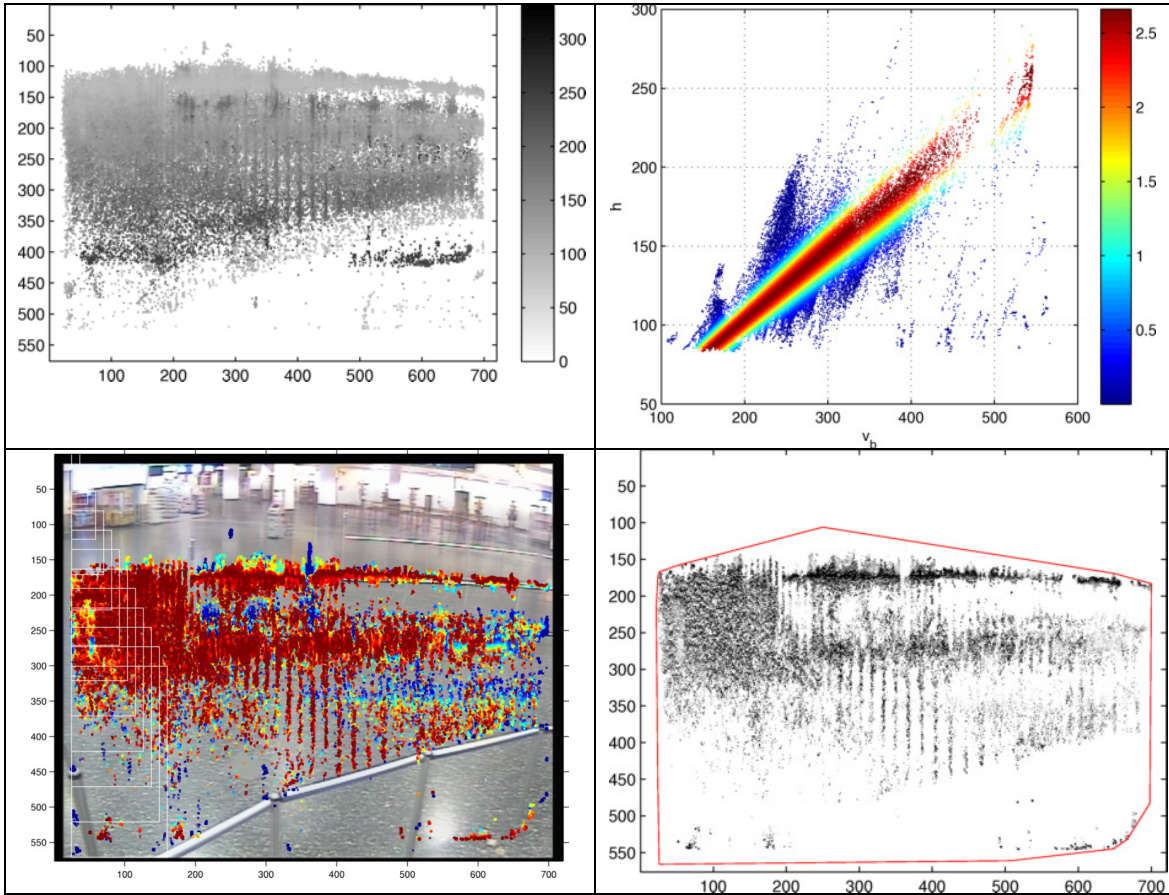
$$h \approx h_c \frac{\Delta y}{y_b - y_0} \quad (1)$$

where  $y_0$  is the row position of the horizon line and  $y_b$  is the bottom image coordinate for the object.

Therefore a simple automatic camera calibration can be performed by regression from all the pedestrian measurements given an average person height  $h_p$  and standard deviation  $\sigma_p$  (here we chose  $h_p = 1.8m$  and  $\sigma_p = 0.15m$ ). The likelihood of the estimated height is assumed Gaussian

$$p(h|h_p, \sigma_p) = \frac{1}{\sqrt{2\pi\sigma_p^2}} e^{-\frac{(h-h_p)^2}{2\sigma_p^2}} \quad (2)$$

Figure 2 gives an example of the object height model derivation for Camera 5. A mask for the ObjectPut event detection is then derived from the convex hull of the pedestrian bottom positions that are in good agreement with the height model.



**Figure 2.** from left to right and top to bottom: average pedestrian height map (dark=tall); object height distribution function of the object vertical position with color function of likelihood (2), most likely object positions with white boxes on the left showing predicted heights by the model; convex hull (in red) of the most likely area for the ObjectPut Detection.

### Object Velocity

Our object tracker was run on the development set. From the object tracks, we derive velocity measurements in pixels/frames. At each image location  $\mathbf{p}$ , we estimated the velocity moments of order  $r$  from all the observed velocities

$$M_r(\mathbf{p}) = \frac{1}{C} \sum_{i=1}^n v_i^r K\left(\frac{\|\mathbf{p} - \mathbf{p}_i\|}{s}\right), \text{ with } C = \sum_{i=1}^n K\left(\frac{\|\mathbf{p} - \mathbf{p}_i\|}{s}\right) \quad (3)$$

where  $K(\cdot)$  is a Gaussian spatial kernel with a spatial width  $s$ . Only moments of order 1 (mean) and 2 are estimated. Those statistics will be used for the “PersonRuns” event detection explained below. On Figure 3 below, we show the resulting statistics for camera 5.

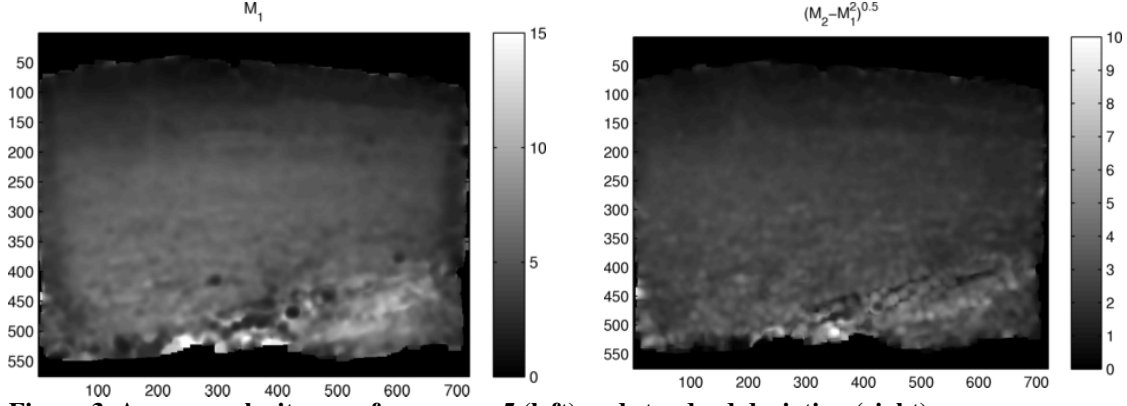


Figure 3. Average velocity map for camera 5 (left) and standard deviation (right).

## II – Person Runs Event

Our tracking system is producing trajectories (tracks) on various detected foreground objects within the scene. For each position in the track we record the object velocity, size and compute the final distance (distance between the track starting point and ending point) as well as the total distance (total distance travelled by the object). The “person running” events were detected by assessing by how much the current velocity diverges from the learned statistics with a kind of a one-sided t-student test. Let’s say that we observe an object with velocity  $v$ , we then compute the Velocity Outlier (VO) score at the confidence level  $\alpha$  based on the learned scene statistics

$$VO_{1-\alpha} = \frac{(v - M_1)}{2t_{\alpha, n-1} \sqrt{M_2 - M_1^2} / \sqrt{n}} \quad (4)$$

where is  $t_{\alpha, n-1}$  is the one sided t-student distribution with  $n-1$  degrees of freedom where  $n$  is the number of observations involved in the computation of  $M_1$  and  $M_2$ . In case of a person running event,  $VO_{1-\alpha}$  should capture the deviation from the mean velocity and take values over 0.5. A confidence level is then computed for each observed track function of the average VO score and the track quality:

$$Conf_{PR} = \min(E\{VO_{\alpha, n-1}\}, 1) \times \frac{\text{Final Distance}}{\text{Total Distance}} \quad (5)$$

Where  $E\{VO_{\alpha, n-1}\}$  is the average outlier score observed over the duration of the track. The ratio of the track Final Distance over Total Distance penalizes tracks that are too noisy.

In order to further reduce the number of false alarms, only events within the learned pedestrian mask (see section I) were processed.

## III – Object Put Event

The “Object Put” detection was based on a very simple dual background model approach described in [5]. Both the long term and short-term background models were Mixture of Gaussians (MoG). The learning rate for the short term was fixed to 1/30 and the long term rate to 1/200. The difference image between the foreground images coming from the two models is then accumulated over time. An example is shown on Figure 4.

In order to reduce false alarms, the object height is validated with the camera geometric model. Only alerts with a height between 10% and 50% of the expected pedestrian height are considered. The confidence level for the event is derived from the average value within the event ROI on the cumulative difference image.



**Figure 4: some frames for an ObjectPut event (left column); foreground difference image (center) and alert images (left)**

#### **IV – Pointing Event**

For the “pointing” event, we implemented an approach based on the learning of compound features proposed recently by Gilbert *et al.* [7][8]. The following steps are involved

1. build an overcomplete set of Harris corners at various spatial scale and in the temporal domain.
2. group corners within a 3x3x3 neighbourhood to form compound features
3. compound features are encoded using information about cell position, scale and corner type to form transactions (or itemsets).
4. a data mining algorithm (APriori algorithm) is applied in order to extract frequent itemsets.
5. transaction rules and associated confidence levels are derived from the frequent itemsets.

The training was very limited due to the lack of time, only the pointing events in the first video of camera 1 in the development set were used. One issue we need to tackle is the large number of transactions generated during the training step (over 1 million transactions were generated here). Another issue is how to take a reliable decision on the presence of Pointing events in a scene where many other actions are taking place (e.g. people walking).

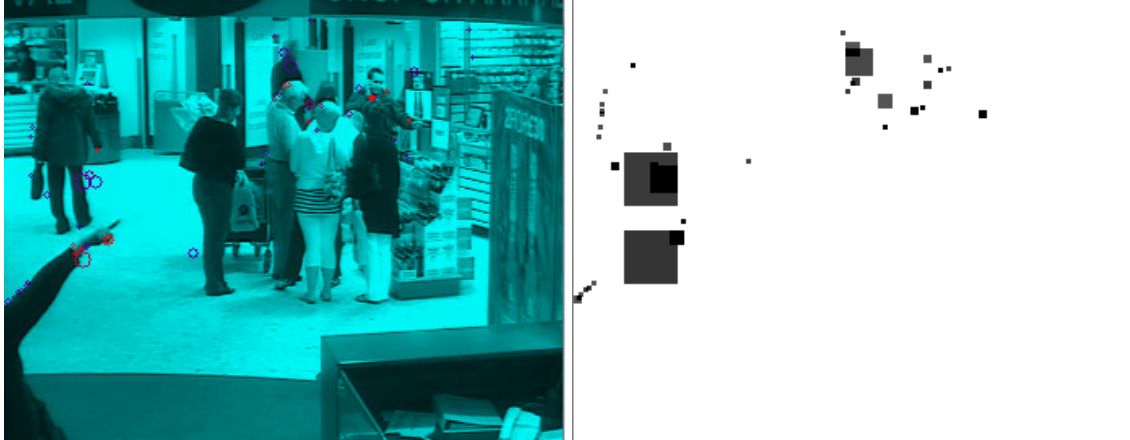


Figure 5: example of a Pointing frame with learned compound features shown as circles (left) and corresponding probability map (right) where dark means high probability.

## V - Results

Overall detection results present too many false alarms especially for PersonRuns and Pointing.

| Event                     | Person Runs | Object Put | Pointing |
|---------------------------|-------------|------------|----------|
| Act. Miss                 | 0.196       | 0.839      | 0.964    |
| Act. RFA (in Events/Hour) | 2110        | 232        | 440      |
| Act. DCR                  | 10.745      | 1.999      | 3.166    |

Table 1: Actual Miss rate and False Alarm rate for each event.

| Event                    | Person Runs | Object Put | Pointing |
|--------------------------|-------------|------------|----------|
| Min Miss                 | 0.944       | 0.955      | 0.988    |
| Min RFA (in Events/Hour) | 68          | 0.394      | 228      |
| Min DCR                  | 1.285       | 0.997      | 2.127    |

Table 2: Minimum Miss rate and False Alarm rate for each event.

## Conclusion

For our first year in this competition, the objective was to put in place our test environment and to be able to deliver results with in-house algorithms that were not necessarily optimal for the TRECVID video corpus. We were not expecting to perform very well as our system was initially designed for an outdoor environment. In particular, our tracker is not performing as it should on this kind of very complex scenes. The current background/foreground segmentation algorithm produces very fragmented blobs and needs to be improved. Also we hope to finish our training for a head detector so that we can track only relevant objects. The method used for Pointing will be further improved in order to handle a larger training set for next year and we are planning to use it for PersonRun and CellToEar.

## Acknowledgments

This work has been supported by MDEIE of the “Gouvernement du Québec”. The authors wish to thank the RQCHP (Compute Canada) for giving access and support to their high performance computing resources.

## References

- [1] S. Foucher, M. Lalonde, L. Gagnon, "Automatic Scene Modeling for Improving Object Classification", SPIE Defense and Security Symposium: Visual Information Processing 2010.
- [2] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection", CVPR 2005.
- [3] D. Hoiem, A.A. Efros, and M. Hebert, "Putting Objects in Perspective", IJCV (80), No. 1, October 2008.
- [4] W. Yang, T. Lan, and G. Mori, "SFU at TRECVID 2009: Event Detection", School of Computer Science, Simon Fraser University.
- [5] Porikli, F., Ivanov, Y. and Haga, T., "Robust Abandoned Object Detection Using Dual Foregrounds", EURASIP Journal on Advances in Signal Processing, 2008.
- [6] P. KadewTraKuPong and R. Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection", Proc. 2nd European Workshop on Advanced Video-Based Surveillance Systems, 2001.
- [7] A Gilbert, J. Illingworth, R. Bowden, "Action Recognition using Mined Hierarchical Compound Features", Accepted for IEEE Trans Pattern Analysis and Machine Learning. 2010.
- [8] A. Gilbert , J. Illingworth, R. Bowden, "Fast Realistic Multi-Action Recognition using Mined Dense Spatio-temporal Features", In Proc. Int. Conference Computer Vision (ICCV09), Kyoto, Japan.