

# Evaluation of Probabilistic Occupancy Map People Detection for Surveillance Systems

Jérôme Berclaz<sup>1</sup> Ali Shahrokni<sup>2</sup> François Fleuret<sup>3</sup> James Ferryman<sup>2</sup> Pascal Fua<sup>1</sup>

<sup>1</sup> CVLab, EPFL, Lausanne, Switzerland

<sup>2</sup> Computational Vision Group, University of Reading, Reading, UK

<sup>3</sup> IDIAP Research Institute, Martigny, Switzerland

jerome.berclaz@epfl.ch, a.shahrokni@reading.ac.uk, francois.fleuret@idiap.ch  
j.m.ferryman@reading.ac.uk, pascal.fua@epfl.ch

## Abstract

In this paper, we evaluate the Probabilistic Occupancy Map (POM) pedestrian detection algorithm on the PETS 2009 benchmark dataset. POM is a multi-camera generative detection method, which estimates ground plane occupancy from multiple background subtraction views. Occupancy probabilities are iteratively estimated by fitting a synthetic model of the background subtraction to the binary foreground motion.

Furthermore, we test the integration of this algorithm into a larger framework designed for understanding human activities in real environments.

We demonstrate accurate detection and localization on the PETS dataset, despite suboptimal calibration and foreground motion segmentation input.

## 1 Introduction

In this paper, we evaluate a state-of-the-art multiple people detection system on the benchmark crowd motion sequences of the PETS 2009 dataset [1]. This dataset contains real life scenarios of crowd activities and provides challenging sequences including occlusions and highly varying illumination conditions in an outdoors environment. The systems we consider have been designed specifically to address the problem of detecting multiple people who occlude each other using a small number of synchronized video sequences such as those depicted in Fig. 1, which represent common task in video-surveillance systems. The core algorithm is based on the Probabilistic Occupancy Map algorithm [7]. It relies on a mathematical framework that robustly estimates the ground plane occupancy at individual time steps. Our framework is capable of accurately localizing individuals not only in camera views but also on the 3D ground plane which is a major advantage compared to other existing systems.



Figure 1: Typical results of the POM algorithm on PETS dataset. Each picture shows a different viewpoint at the same time frame.

Furthermore, we integrate the POM algorithm in a Scene Understanding System (SUS) designed for understanding human activities in real environments [9]. The aim of the SUS framework is to identify objects and events, and extract sense from scene observation. In this framework, we employ a multi-camera system that handles the processing tasks including foreground detection and target tracking.

In the detection stage, input images are processed as they arrive from the camera to locate objects that are of interest. At the most basic level this is performed by assuming that the objects of interest are moving and can thus be segmented by detecting motion in the images. For a surveillance system, this is typically a reasonable assumption, as objects of interest will never remain static for the duration of the tracking scenario. Among different available background subtraction techniques implemented on the system, we use color mean and variance-based blob detection tech-

nique [11] to classify pixels in the input sequences as foreground, background, “shaded background” or “highlighted background”. This method is shown to be robust to shadows or highlights. The background model is updated at each frame to handle changes in illumination by incorporating part of the new pixel values, determined by the learning rate.

The segmented binary foreground images are then passed to POM for people detection which are subsequently used for higher level recognition. Finally, within the SUS framework, 3D localization and data fusion stages then combine the results from multiple cameras into a coherent interpretation of the scene, cameras and the detected objects.

In the remainder of this paper, we describe the POM algorithm for people detection in Section 3. We present qualitative analysis of the tracking results in Section 4. We compare results of the stand alone POM algorithm and when combined with the detection stage of the SUS framework described above. Finally the conclusion and future work is discussed in Section 5.

## 2 Related Work

State-of-the-art approaches to people detection can be divided into monocular and multi-view methods.

### 2.1 Monocular

Among monocular approaches, an important class relies on blobs created by background subtraction [12, 4, 10]. Such a technique works well for fixed cameras, as long as the number of targets is low and that people do not occlude each other too much.

To overcome some of the limitations of the blob-based methods, another class of work tackles the people detection problem using classification [20, 17, 5, 15, 6]. These approaches have a higher discrimination power than the blob-based and can deal better with occlusions and dense crowds. They also do not require a static camera. However, they usually need a careful training phase and are prone to confusion with objects that exhibit a human-like shape.

### 2.2 Multi-view

Despite the effectiveness of monocular methods, the use of multiple cameras is necessary when dealing with a higher number of people, which usually produces many occlusions. A number of approaches [3, 8, 19] rely on extracting foreground motion blobs, that are fused in various ways from the multiple available views.

Other approaches use color information [16, 13] or classification [2] to obtain evidence about people presence in



Figure 2: The ground plane grid used for the PETS sequence. Note that people moving outside this grid will not be detected by the system.

single view prior to merging the information from multiple views.

By contrast, the POM detector uses a sophisticated approach based on a generative model to estimate the probabilities of occupancy, and explicitly handles complex occlusion interactions between detected individuals

## 3 Probabilistic Occupancy Map

We describe here the main algorithm used for people detection. It is a multi-camera generative method that iteratively estimates probabilities of occupancy in the ground plane.

More specifically, the monitored area is divided into a grid of  $G$  locations, each of which a square of about  $25 \times 25$  cm, corresponding roughly to the space occupied by a standing pedestrian as illustrated in Fig. 2. Images from all  $C$  cameras of the system are individually processed by background subtraction, yielding binary images with motion blobs, such as those shown in Fig. 3.

The cameras used by the system are fixed and calibrated. Using this calibration, we compute homographies mapping the ground plane from the camera views to the same plane seen from a virtual top view. Such a model makes it possible to project in camera views, for a given location  $k$  of the ground plane, a rectangle of roughly the same size and aspect ratio as the motion blob that a pedestrian standing at location  $k$  would produce after background subtraction. This method allows us to establish correspondences between the cameras and the top view. Examples are shown in Fig. 4.

At the core of our approach is an iterative procedure for estimating occupancy probabilities associated with every location of the ground plane. At each iteration, occupancy

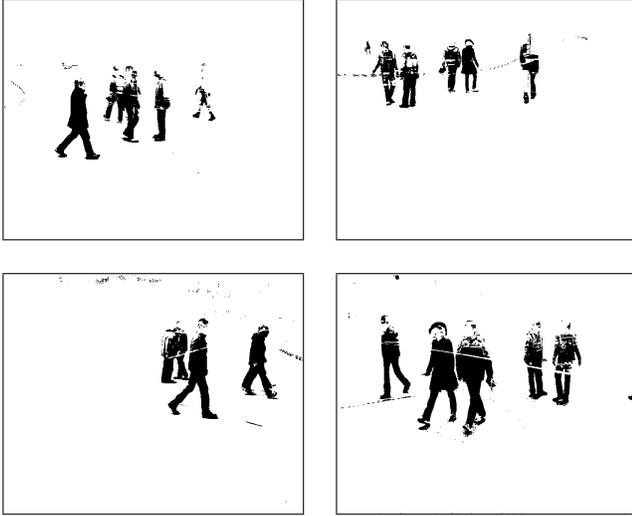


Figure 3: Typical input for POM algorithm: several camera views at the same time frame, processed by background subtraction.

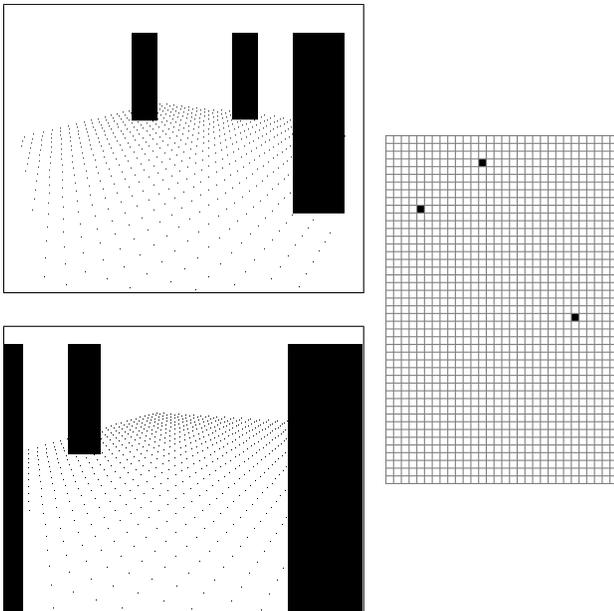


Figure 4: An example of camera calibration. The two left images are synthetic views of two different cameras, corresponding to the ground plane occupancy shown by the top view on the right.

probabilities for every ground location are re-estimated, in order to minimize the difference between the corresponding synthetic images, generated with projected rectangles and the real background subtraction images.

Let us consider a discretization of the ground plane in  $G$  locations. For every location  $k \in [1, G]$ , we define  $X_k$  as the Boolean random variable standing for the occupancy of the location. We call  $\mathbf{X} = [X_1, \dots, X_G]$  the binary images produced by the background subtraction for all  $C$  cameras of the system. Our goal is to estimate the occupancy of every ground plane location, given the background subtraction images

$$P(\mathbf{X} | \mathbf{B}), \quad (1)$$

where  $\mathbf{X} = X_1, \dots, X_G$ . Using the Bayes formula we can rewrite (1) as:

$$P(\mathbf{X} | \mathbf{B}) = \frac{P(\mathbf{B} | \mathbf{X}) P(\mathbf{X})}{P(\mathbf{B})}. \quad (2)$$

Two independence assumptions are needed to derive the update equation of Section 3.2. First of all, we consider the occupancy of a location independent from the other locations of the grid. This amounts to ignoring social conventions, such as the tendency by people of not standing closer than a given distance to each other, or the fact that people are more likely to walk in groups than alone. Avoidance strategies by people are also ignored. This first independence assumption allows us to write

$$P(\mathbf{X}) = P(X_1, \dots, X_G) = \prod_k P(X_k). \quad (3)$$

The second assumption is that all statistical dependencies between background subtraction views originate from the presence of people. In other words, various background subtraction views from different cameras are independent from each other, given the true occupancy of the ground plane:

$$P(B_1, \dots, B_C | \mathbf{X}) = \prod_c P(B_c | \mathbf{X}). \quad (4)$$

### 3.1 Generative Model

Here we describe the generative model  $P(B_c | \mathbf{X})$  of the background subtraction image given the occupancy of the ground plane. We call  $A_c$  the synthetic image in which, using camera calibration, we put rectangles at the location where people are expected. Examples of such images are shown in Fig. 4. Since these images are a function of the ground plan occupancy  $\mathbf{X}$ , they are themselves a random quantity. We model the background subtraction images  $B_c$  as if it was the synthetic image with some random noise.

To compare the similarity of a background subtraction image  $B_c$  with its synthetic counterpart  $A_c$ , we define a pseudo-distance functions between images  $\Psi(A, B)$ , which is roughly the normalized intersection of the pixels of both images  $A$  and  $B$  (see [7] for details). With this definition in mind, we model the conditional probability  $P(\mathbf{B}|\mathbf{X})$  of the background subtraction images given the occupancy state of the ground plane, with a density decreasing with the pseudo distance between  $B_c$  and  $A_c$ :

$$\begin{aligned} P(\mathbf{B}|\mathbf{X}) &= \prod_c P(B_c|\mathbf{X}) \\ &= \prod_c P(B_c|A_c) \\ &= \frac{1}{Z} \prod_c e^{-\Psi(B_c, A_c)}. \end{aligned} \quad (5)$$

### 3.2 Probabilities Estimation

We approximate the conditional occupancy probability of the ground plane given the background subtraction images  $P(\mathbf{X}|\mathbf{B})$  with a product law  $Q$ , and denote by  $q_1, \dots, q_G$  its marginal probabilities, which in turn approximate  $P(X_1|\mathbf{B}), \dots, P(X_G|\mathbf{B})$ . Our goal is thus to find an approximation  $Q$  as close as possible to the real distribution  $P(\cdot|\mathbf{B})$ . Therefore, we want to minimize the Kullback-Leibler divergence between the two distributions

$$KL(Q, P(\cdot|\mathbf{B})). \quad (6)$$

The minimization of (6) leads to

$$q_k = 1/(1 + \exp(\lambda_k + \sum_c E_Q(\Psi(B_c, A_c) | X_k = 1) - E_Q(\Psi(B_c, A_c) | X_k = 0))), \quad (7)$$

where  $E_Q$  denotes the expectation under  $\mathbf{X} \sim Q$ ,  $\lambda_k = \log \frac{1-\epsilon_k}{\epsilon_k}$ , and  $\epsilon_k$  is the prior probability of presence at location  $k$ , i.e.  $\epsilon_k = P(X_k = 1)$ .

The computation of  $E_Q(\Psi(B_c, A_c)|X_k = \xi)$  is however intractable. We use the fact that, under  $\mathbf{X} \sim Q$ , the image  $A_c$  is concentrated around  $B_c$ , to further approximate

$$E_Q(\Psi(B_c, A_c)|X_k = \xi) \simeq \Psi(B_c, E_Q(A_c|X_k = \xi)). \quad (8)$$

This yields our final update equation

$$q_k = 1/(1 + \exp(\lambda_k + \sum_c \Psi(B_c, E_Q(A_c|X_k = 1)) - \Psi(B_c, E_Q(A_c|X_k = 0)))). \quad (9)$$

Note that  $E_Q(A_c)$  is the expectation of the synthetic image of camera  $c$ , under the probability distribution  $Q$ . We call this quantity *synthetic average image*.  $E_Q(A_c|X_k = 0)$  and  $E_Q(A_c|X_k = 1)$  are synthetic average images equal to  $E_Q(A_c)$ , with  $q_k$  forced to 0, or 1 respectively. The derivation of Eq. 7 is explained in details in [7].

### 3.3 Iterative Algorithm

The final algorithm works as follows. All the  $q_k$  are first given a uniform prior value,  $\epsilon$ . They are then iteratively updated with Eq. 9, using the estimate computed at the previous iteration. This scheme converges after a number of iterations, which is usually of the order of 100. Fig. 5 illustrates this with a few average synthetic images of the probability estimates, taken at various steps of the convergence process.

## 4 Experimental Results

We tested the POM algorithm on the multi-camera sequence S2-L1 from the PETS 2009 data set. In this video, 7 cameras observe several pedestrians under various angles. Among the cameras, 4 of them are located relatively close to the scene, and at roughly the same height as people's head. The 3 remaining cameras are located further from the monitored area and about 4-5m above the ground, giving a wider angle view of the situation.

### 4.1 Background Subtraction

The background subtraction algorithm used by the stand alone version of POM is our own implementation of the *Eigenbackground* method [18]. It typically produces results such as those illustrated in Fig. 3. We generated the background model using some images of the dataset sequence S0, featuring empty background. Important lighting changes occurring between the sequence and the reference images used for the background model impair the quality of background subtraction on certain views, as shown in Fig. 6-a.

### 4.2 Camera Calibration

For the camera calibration, we used the calibration provided by PETS and adapted it to obtain the homographies needed by our model. Our tests revealed that the correspondences across cameras were not as accurate as required by our algorithm, resulting in some bounding boxes being offset by more than one grid location in some areas of certain camera views. An example of this imprecision is shown in Fig. 6-b. We suspect that this imprecision stems from the fact that we

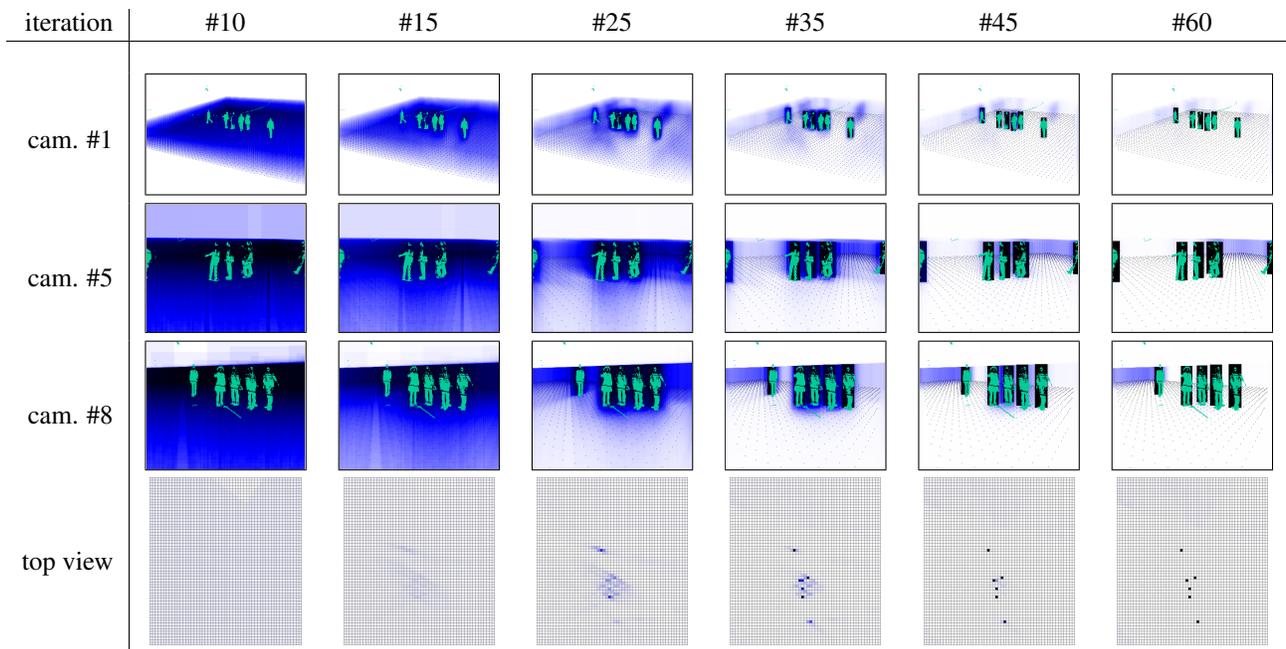


Figure 5: POM’s iterative process convergence on a single time frame. Each of the first three rows shows a different camera view, while the last one displays a top view. Every column shows the convergence process at a different iteration. The green shapes are the motion blobs computed by background subtraction. This is the actual input for POM. The shade of blue represent the probability of occupancy, the darker the higher probability. This figure is best viewed in color.

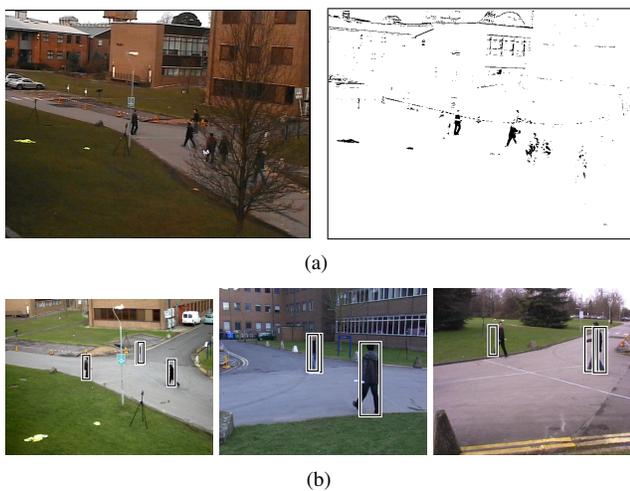


Figure 6: Some of the difficulties encountered in the PETS sequences. (a) shows suboptimal background subtraction results due to poor illumination, and (b) illustrates calibration inaccuracy: The two left views show correct bounding box alignment, whereas the rightmost view is clearly misaligned.

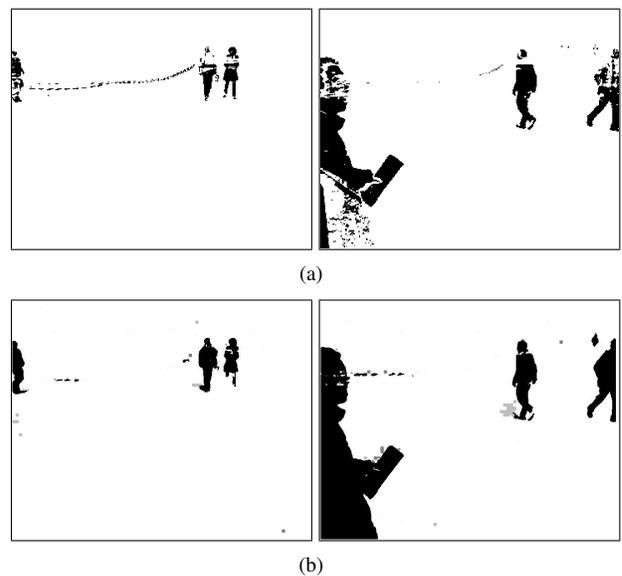


Figure 7: A comparison between the background subtraction based on Eigenbackground (a), as used in the stand alone POM, and the color mean and variance-based one (b), used in the integrated people detector.

are modeling the ground as a plane, whereas in this case it is rather uneven. For this reason, we decided to use only 5 out of the 7 available views, discarding the 2 views with less precise calibration data.

We defined the area covered by our system to the  $18\text{m} \times 20\text{m}$  rectangle shown in Fig. 2. This space was discretized into  $55 \times 61 = 3,355$  locations.

### 4.3 Stand Alone Results

The quantitative results of the PETS evaluation are shown in Table 1, whereas some typical detection results are illustrated in Fig. 8. Two metrics are used, namely *Multiple Object Detection Accuracy* (MODA) and *Multiple Object Detection Precision* (MODP). As suggested by their names, the first one accounts for the accuracy of the detection and the second one for its precision. Both metrics are described in [14].

Considering the difficulties presented by the sequence, among which the imprecision of the calibration and some low quality background subtraction, the results we obtain are satisfactory. As illustrated by the detection results of Fig. 8, the localization precision is generally high. Note that POM does not only detect people in the camera views, but also accurately localize them on the 3D ground plane which is a major advantage of our system compared to existing methods. Not surprisingly, the two camera views that we did not use for detection because of calibration issues - namely cameras 2 and 7 - have by far the worst results of the whole set. The results on the other cameras are quite uniform.

The choice of using fewer views than available is a trade-off: It allows us to gain precision, but at the cost of reducing the coverage of some areas, away from the center. As a result, some people are not detected when moving at the border of the grid.

### 4.4 Integrated Results

We present results obtained on the same sequence using the integrated version of POM and SUS. The main difference here is that the input to POM comes from a more elaborate background subtraction algorithm [9]. A qualitative comparison between the background subtraction results from the two methods is given in Fig. 7.

Although the integrated background subtraction blobs are visually more accurate, the final detection results are of the overall same quality as those from the stand alone POM, as seen on Table 2. This can be explained by the fact that POM approximates human-shaped motion blobs by a crude rectangle, and is therefore not sensitive at all to small shape variations in motion blobs. As for the stand alone version, the scores for all camera views are quite uniform, except

for the two misaligned cameras, whose input was not used for detection. A few detection results of this method are displayed in the last three rows of Fig. 8.

## 5 Conclusion

We have presented the results of the POM people detector on the PETS 2009 dataset. We have shown its effectiveness in accurately detecting multiple people in outdoors environments including occlusions and highly varying illumination conditions. Our framework is capable of accurately localizing individuals not only in camera views but also on the 3D ground plane, which is a major advantage compared to other existing systems. Segmented binary foreground images are used as input to the POM algorithm. In spite of the crude nature of the input to the algorithm, we can successfully track people on ground plane and estimate a probabilistic occupancy map. Furthermore, we have integrated the POM algorithm in a multi-camera scene understanding framework designed for understanding human activities in real environments. We provide qualitative results as well as the submitted detection details using both stand alone and the integrated version of POM algorithm on dataset S2.L1 of PETS 2009. We further plan to use POM algorithm to estimate crowd density on other PETS 2009 sequences.

## References

- [1] Eleventh IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, June 2009. <http://pets2009.net>.
- [2] J. Berclaz, F. Fleuret, and P. Fua. Principled Detection-by-Classification from Multiple Views. In *International Conference on Computer Vision Theory and Applications*, Lisbon, Portugal, January 2008.
- [3] J. Black, T.J. Ellis, and P. Rosin. Multi-view image surveillance and tracking. In *IEEE Workshop on Motion and Video Computing*, 2002.
- [4] Robert T. Collins. Mean-shift blob tracking through scale space. In *Conference on Computer Vision and Pattern Recognition*, volume 02, page 234, Los Alamitos, CA, USA, 2003. IEEE Computer Society.
- [5] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *Conference on Computer Vision and Pattern Recognition*, 2005.
- [6] Andreas Ess, Bastian Leibe, Konrad Schindler, and Luc Van Gool. A mobile vision system for robust multi-person tracking. In *Conference on Computer Vision and Pattern Recognition*, 2008.

metric	Camera 1	Camera 2	Camera 3	Camera 4	Camera 5	Camera 6	Camera 7
MODP	0.65	0.29	0.65	0.6	0.57	0.65	0.46
MODA	0.67	-0.72	0.64	0.61	0.72	0.7	0.67

Table 1: Evaluation results for the stand-alone POM algorithm, using *Multiple Object Detection Accuracy* (MODA) and *Multiple Object Detection Precision* (MODP) metrics.

metric	Camera 1	Camera 2	Camera 3	Camera 4	Camera 5	Camera 6	Camera 7
MODP	0.65	0.46	0.62	0.56	0.58	0.65	0.47
MODA	0.6	0.59	0.58	0.54	0.65	0.75	0.58

Table 2: Evaluation results for the integrated version of the POM algorithm, using *Multiple Object Detection Accuracy* (MODA) and *Multiple Object Detection Precision* (MODP) metrics.

- [7] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multi-Camera People Tracking with a Probabilistic Occupancy Map. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):267–282, February 2008.
- [8] D. Focken and R. Stiefelhagen. Towards vision-based 3d people tracking in a smart room. In *IEEE International Conference on Multimodal Interfaces*, 2002.
- [9] F. Fusier, V. Valentin, F. Brémond, M. Thonnat, M. Borg, D. Thirde, and J. Ferryman. Video understanding for complex activity recognition. *Machine Vision and Applications Journal*, 18:167–188, 2007.
- [10] Mei Han, Wei Xu, Hai Tao, and Yihong Gong. An algorithm for multiple object trajectory tracking. In *Conference on Computer Vision and Pattern Recognition*, volume 1, pages 864–871, June 2004.
- [11] T. Horprasert, D. Harwood, and L.S. Davis. A statistical approach for real-time robust background subtraction and shadow detection. *Proc. IEEE ICCV’99 FRAME-RATE Workshop*, 1999.
- [12] M. Isard and J. MacCormick. BraMBLe: a Bayesian multiple-blob tracker. In *Conference on Computer Vision and Pattern Recognition*, volume 2, pages 34–41, July 2001.
- [13] J. Kang, I. Cohen, and G. Medioni. Tracking people in crowded scenes across multiple cameras. In *Asian Conference on Computer Vision*, 2004.
- [14] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, and Jing Zhang. Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(2):319–336, Feb. 2009.
- [15] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *Conference on Computer Vision and Pattern Recognition*, volume 1, San Diego, CA, June 2005.
- [16] A. Mittal and L. Davis. M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. *International Journal of Computer Vision*, 51(3):189–203, 2003.
- [17] K. Okuma, A. Taleghani, N. de Freitas, J.J. Little, and D.G. Lowe. A boosted particle filter: multitarget detection and tracking. In *ECCV*, Prague, Czech Republic, May 2004.
- [18] Nuria M. Oliver, Barbara Rosario, and Alex P. Pentland. A bayesian computer vision system for modeling human interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):831–843, 2000.
- [19] K. Otsuka and N. Mukawa. Multi-view occlusion analysis for tracking densely populated objects based on 2-d visual angles. In *Conference on Computer Vision and Pattern Recognition*, 2004.
- [20] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *International Conference on Computer Vision*, pages 734–741, 2003.

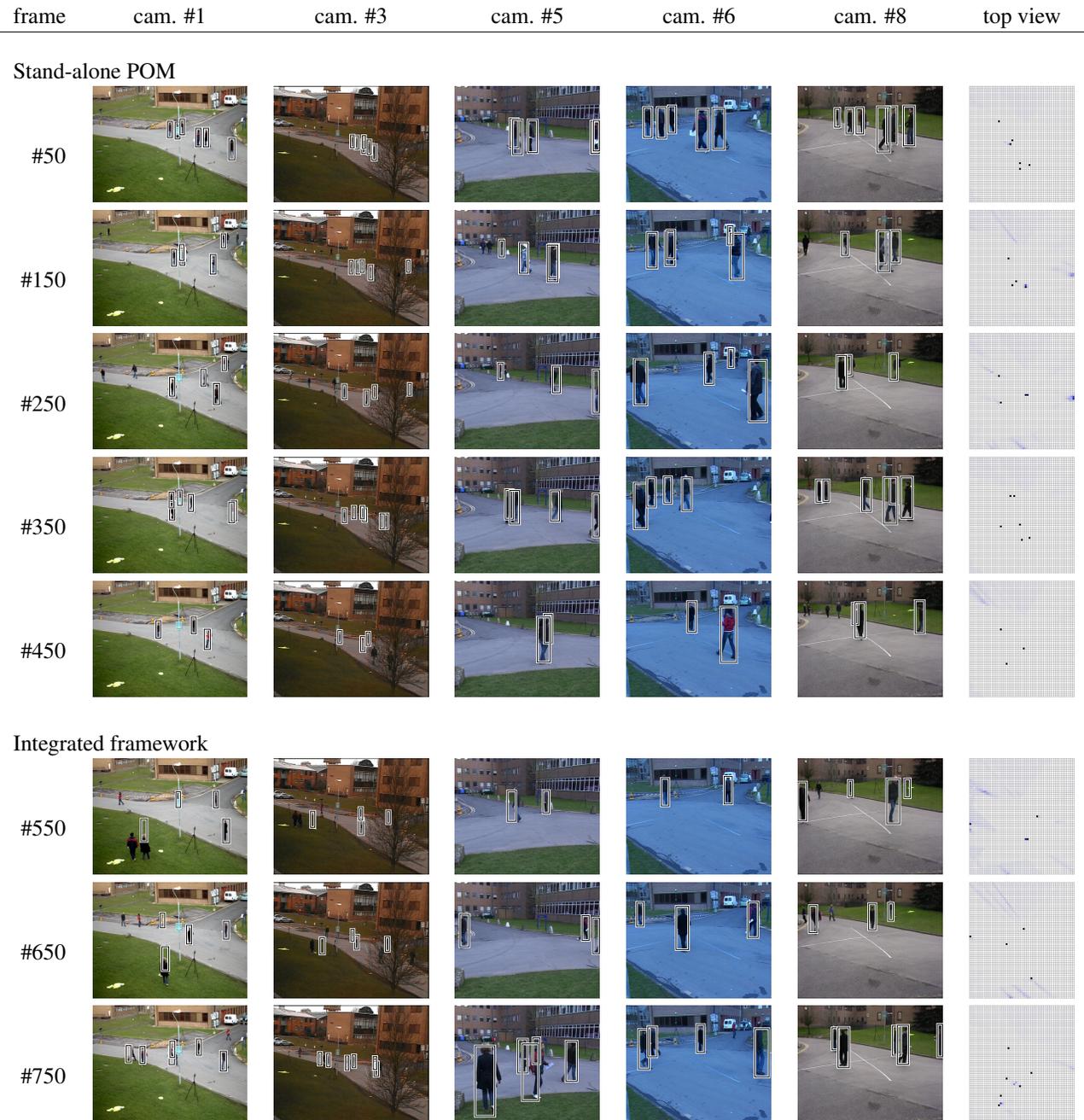


Figure 8: Some detection results of POM on the PETS 2009 dataset. Each row shows a different time frame of the video sequence. The last column shows the probabilities of occupancy in top view. Those probabilities have been thresholded to yield the detections shown in the camera views. The first five rows show results from the stand-alone POM and the last three rows from the integrated framework.