DALES: Automated Tool for Detection, Annotation, Labelling and Segmentation of Multiple Objects in Multi-Camera Video Streams

M. Bhat and J. I. Olszewska
University of Gloucestershire
School of Computing and Technology
The Park, Cheltenham, GL50 2RH, UK
jolszewska@glos.ac.uk

Abstract

In this paper, we propose a new software tool called DALES to extract semantic information from multi-view videos based on the analysis of their visual content. Our system is fully automatic and is well suited for multi-camera environment. Once the multi-view video sequences are loaded into DALES, our software performs the detection, counting, and segmentation of the visual objects evolving in the provided video streams. Then, these objects of interest are processed in order to be labelled, and the related frames are thus annotated with the corresponding semantic content. Moreover, a textual script is automatically generated with the video annotations. DALES system shows excellent performance in terms of accuracy and computational speed and is robustly designed to ensure view synchronization.

1 Introduction

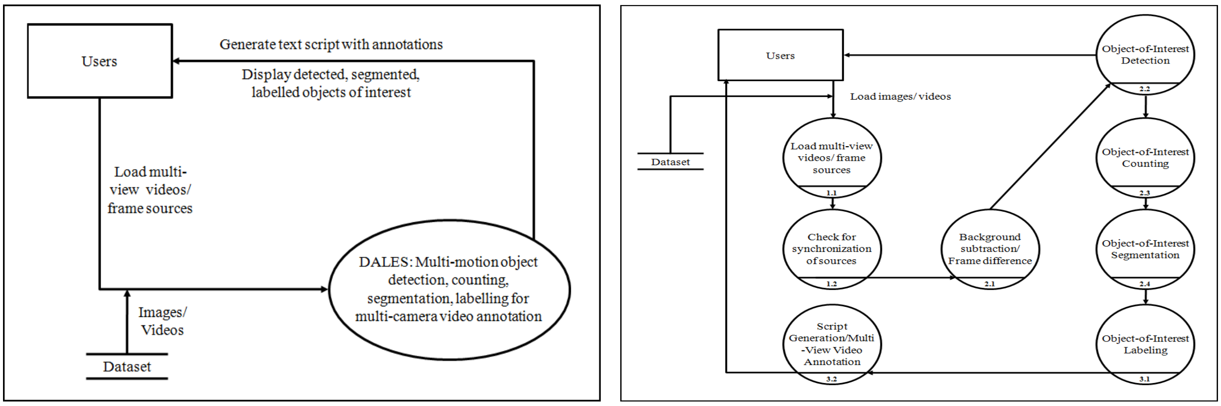
With the increasing use of electronic equipments, storage devices and computational systems for applications such as video surveillance (Kumar et al., 2010) and sport event monitoring (Alsuqayhi and Olszewska, 2013), the development of automated tools to process the resulting big amount of visual data in order to extract meaningful information becomes a necessity.

In particular, the design of multi-view video annotation systems is a challenging, new task. It aims to process multi-view video streams which consist of video sequences of a dynamic scene captured simultaneously by multiple cameras. Such multi-input system is dedicated to automatically analyse the visual content of the multi-camera records and to generate semantic and visual annotations, in the way to assist users in the understanding and reasoning about large amount of acquired data.

For this purpose, data should be processed through different, major stages such as object-of-interest detection and segmentation, frame labelling, and video annotation. In the literature, most of the works dealing with the analysis of multi-camera video streams are focused on the sole task of tracking multiple, moving objects and use different approaches such as background subtraction (Diaz et al., 2013), Bayesian framework (Hsu et al., 2013), particle filter (Choi and Yoo, 2013), or Cardinalized Probability Hypothesis Density (CPHD) based filter (Lamard et al., 2013). On the other hand, research on video annotation has lead to the development of several efficient systems (Town, 2004; Natarajan and Nevatia, 2005; Bai et al., 2007; Vrusias et al., 2007), but all designed for a single camera video stream input.

In this paper, we describe a full system which takes multi-camera video stream inputs and performs visual data processing to generate multi-view video annotations. Our system has been developed in context of outdoor video-surveillance and is an automatic Detection, Annotation, LabElling and Segmentation (DALES) software tool. It presents also the advantage to have an entire chain of data processing from visual to textual one, reducing thus the semantic gap.

As camera calibration is in general an expensive process (Black et al., 2002) and in real life, surveillance application measurements of camera parameters are not readily available (Guler et al., 2003), DALES system does not involve any camera calibration parameters.


Users
Generate text script with annotations
Display detected, segmented, labelled objects of interest
Load multi-view videos/ frame sources
DALES: Multi-motion object detection, counting, segmentation, labelling for multi-camera video annotation
Images/ Videos
Dataset
Users
Load images/ videos
Dataset
Load multi-view videos/ frame sources
1.1
Check for synchronization of sources
1.2
Background subtraction/ Frame difference
2.1
Object-of-Interest Detection
2.2
Object-of-Interest Counting
2.3
Object-of-Interest Segmentation
2.4
Object-of-Interest Labeling
3.1
Script Generation/Multi -View Video Annotation
3.2

(a)

(b)

Figure 1: Overview of the flow of data within DALES software: (a) Context level Data Flow Diagram; (b) Second level Data Flow Diagram.

DALES provides not only the automatic annotation of multi-view video streams, but also performs, in multi-camera environment, tasks such as multiple object-of-interest detection and segmentation, target counting and labelling, and image annotations. Our approach could cope with any dynamic scene recorded by pan-tilt-zoom (PTZ) static cameras with overlapping color views, unlike systems such as (Kettnaker and Zabih, 1999) based on non-overlapping cameras. The acquired multi-view sequences could contain complex backgrounds, moving objects or noisy foregrounds, and present illumination variations or poor resolution.

Hence, the contribution of this paper is twofold:

- the automated, textual annotation of multi-camera video streams based on visual features;
- the development of a full, automatic system covering all the phases from multiple, visual multi-motion target detection to multi-view video annotation and text-script generation.

The paper is structured as follows. In Section 2, we describe our DALES system for fast, multiple video-object detection, segmentation, labeling and effective multi-view video annotation. Our tool has been successfully tested on standard, real-world video-surveillance dataset as reported and discussed in Section 3. Conclusions are presented in Section 4.

2 DALES System

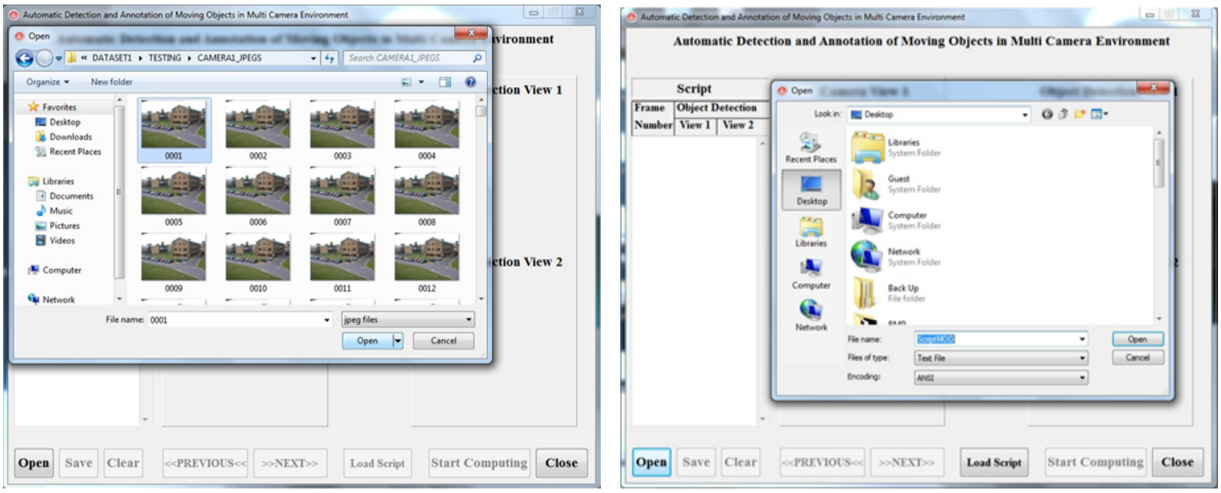
DALES system architecture is presented in Section 2.1, while its two main computational stages, one at the object-of-interest level, the second one at the frame/video levels are described in Sections 2.2-2.3 and Section 2.4, respectively.

2.1 System Architecture

DALES software tool has been prototyped according to the Rapid Application Development (RAD) methodology and using object-oriented approach (C++, 2011). The data flow diagrams (DFDs) shown in Figs. 1 (a)-(b) display the flow of data within different stages of the system. DFDs give an account of the type of data input, the processing involved with these data as well as the final data we get, each higher level of DFDs elaborating the system further.

In order to start the computation of multi-view, multiple object-of-interest detection and counting, target segmentation and labelling, multi-stream video annotations and script generation, DALES system requires multi-view camera synchronization. This is achieved in the multiple views by checking the correspondence of the file names of the files in all the views during the loading phase (Figs. 2(a), 3(a)).

On the other hand, DALES could be used as a viewer of annotated, multi-view videos by loading the generated text script (Fig. 2(b)).



(a)

(b)

Figure 2: The initialisation of DALES software could be done either by providing (a) the video sequence folder name, in order to use the software to process the multi-view video; or by providing (b) a text script with video annotation, in order to run the software to visualize the frames and corresponding annotations.

2.2 Multiple-Object Detection, Segmentation, and Counting

A number of solutions exists for detecting multiple objects of interest in video scenes (Olszewska and McCluskey, 2011; Olszewska, 2011; Olszewska, 2012b). In particular, background subtraction has long been used in the literature as it provides a useful approach to both detect and segment objects of interest. This method could be computed by difference between two consecutive frames (Archetti et al., 2006), by subtracting the current frame from the background (Toyama et al., 1995; Haritaoglu et al., 2000), or combining both frame difference and background subtraction techniques (Huang et al., 2007; Yao et al., 2009).

In our scenario, images $I(x,y)$ we consider may be taken from very different cameras, different lighting, etc. For that, we compute blobs separately in each of the views. The blobs are defined by labeled connected regions, which are obtained by background subtraction. The latter technique consists in computing the difference between the current image intensity $I(x,y)$ and a background model, and afterwards, in extracting the foreground.

To model the background, we adopt the running Gaussian average (RGA) (Wren et al., 1997), characterized by the mean μ_b and the variance σ_b^2, rather than, for example, the Gaussian mixture model (GMM) (Stauffer and Grimson, 1999; Friedman and Russell, 1997; Zivkovic and van der Heijden, 2004), since the RGA method is much more suitable for real-time tracking.

Next, the foreground is determined by

$$F(x,y) = \begin{cases} 1 & \text{if } |I(x,y) - \mu_b| > n \cdot \sigma_b, \text{ with } n \in \mathbb{N}_0, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Finally, morphological operations (Haralick, 1988) are applied to the extracted foreground F, in order to exploit the existing information on the neighboring pixels,

$$f(x,y) = Morph(F(x,y)). \tag{2}$$

2.3 Object Labeling

In this work, the detected and segmented object from a frame is automatically labelled by comparing it with a given example object based on the Scale Invariant Feature Transform (SIFT) descriptors (Olszewska, 2012a).

Algorithm 1 Matching Algorithm

```
Given $A' = A, B' = B, M = \varnothing$,
for all $a_i \in A'$ do
    for all $b_j \in B'$ do
        repeat
            if
                $d_P(a_i,b_j) = \min_{b \in B'} d_P(a_i,b)$
                $\wedge \quad d_P(b_j,a_i) = \min_{a \in A'} d_P(b_j,a)$
                $\wedge \quad d_P(a_i,b_j) \leq d_H(A,B)$
                $\wedge \quad d_P(b_j,a_i) \leq d_H(A,B)$
            then
                $(a_i,b_j) \subset M$
                $\wedge \quad A' = A \backslash \{a_i\} \wedge B' = B \backslash \{b_j\}$
            end if
        until $A' \neq \varnothing \vee B' \neq \varnothing$
    end for
end for
return $M$
```

The main steps of our labelling process are (i) the detection of object's SIFT features which are robust to rotation, translation, scale changes as well as some viewpoint variations, (ii) their matching by means of the Embedded Double Matching Algorithm (Algorithm 1) and (iii) the label inheritance. Moreover, our approach does not require any training and thus is online compatible.

The comparison between the query object and the candidate one is performed in the feature space in order to be more computationally effective and to be more robust towards noise and affine transformation, and is followed by the computation of an associated similarity measure $d_S(A,B)$, which is computed as follows

$$d_S(A,B) = \frac{\#M}{\frac{\#A + \#B}{2}}, \tag{3}$$

with A and B, the sets of SIFT features of the query and the candidate objects, respectively, and M, the set of the double-matched ones (Alqaisi et al., 2012).

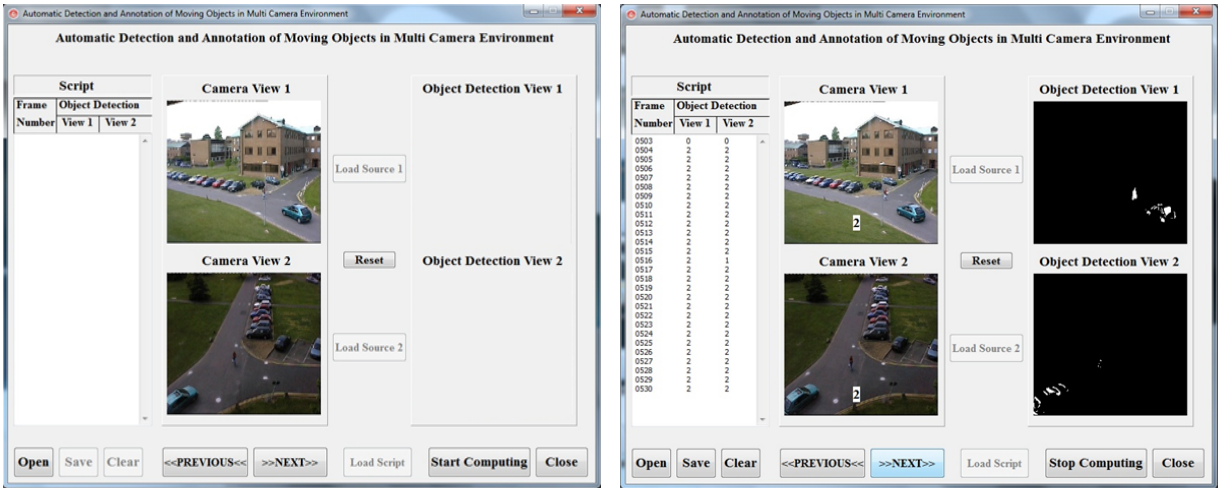
The decision that a candidate object contains similar content to the query one is taken based on the fact that the similarity measure $d_S(A,B)$ is above a given threshold. In the case when the similarity measure $d_S(A,B)$ is below the given threshold, the candidate is rejected.

Finally, once the decision that a candidate object contains similar content to the query one has been taken, the label of the candidate object is automatically mapped with the predefined label of the example object.

2.4 Multi-View Annotations

Hence, by using different objects' examples, all the frames from the video dataset are indexed with the relevant semantic labels based on their visual content similarity, while the objects of interest are automatically labeled and localized within these frames (Olszewska, 2012a).

Unlike (Evans et al., 2013), which uses an early fusion, where all the cameras are used to make a decision about detection and tracking of the objects of interest, DALES system performs a late fusion. Indeed, in our system, objects of interest are detected and labelled in individual cameras independently. Next, the results are combined on the majority voting principle based on the semantic consistency of



Frame Number	Object Detection	
	View 1	View 2
0503	0	0
0504	2	2
0505	2	2
0506	2	2
0507	2	2
0508	2	2
0509	2	2
0510	2	2
0511	2	2
0512	2	2
0513	2	2
0514	2	2
0515	2	2
0516	2	1
0517	2	2
0518	2	2
0519	2	2
0520	2	2
0521	2	2
0522	2	2
0523	2	2
0524	2	2
0525	2	2
0526	2	2
0527	2	2
0528	2	2
0529	2	2
0530	2	2

(a)

(b)

Figure 3: Snapshots of DALES software windows, in the phase of: (a) loaded multi views; (b) detected moving objects.

the labels in sequential frames and across multiple camera views, rather than exploring geometrical correspondences of objects as in (Dai and Payandeh, 2013).

3 Experiments and Discussion

To validate the DALES tool, we have applied our system on the standard dataset (PETS, 2001) consisting of video-surveillance dynamic scene recorded by two PTZ cameras. This produces two videos. Each contains 2688 frames, whose average resolution is of 576x768 pixels and which were captured in outdoor environment. This database owns challenges of multi-view video stream, as well as quantity, pose, motion, size, appearance and scale variations of the objects of interest, i.e. people and cars.

All the experiments have been run on a computer with Intel Core 2 Duo Pentium T9300, 2.5 GHz, 2Gb RAM, and using our DALES software implemented with C++ (C++, 2011).

Some examples of the results of our DALES system are presented in Fig. 3(b). These frames present difficult situations such as poor foreground/background contrast or light reflection.

To assess the detection accuracy of DALES system, we adopt the standard criteria (Izadi and Saeedi, 2008) as follows:

$$detection\ rate\ (DR) = \frac{TP}{TP+FN}, \tag{4}$$

$$false\ detection\ rate\ (FAR) = \frac{FP}{FP+TP}, \tag{5}$$

with TP, true positive, FP, false positive, and FN, false negative.

The labelling accuracy of DALES system could be assessed using the following standard criterion:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN}, \tag{6}$$

with TN, true negative.

In Table 1, we have reported the average detection and false alarm rates of our DALES method against the rates achieved by (Izadi and Saeedi, 2008), while in Table 2, we have displayed the average accuracy of object-of-interest labelling of our DALES method against the rate obtained by (Athanasiadis et al., 2007).

Table 1: Average detection rates of object-of-interests in video frames.

	(Izadi and Saeedi, 2008)	**DALES**
average detection rate (DR)	91.3%	91.6%
average false alarm rate (FAR)	9.5%	4.9%

Table 2: Average accuracy of object-of-interest labelling in video frames.

	(Athanasiadis et al., 2007)	**DALES**
average accuracy	85%	95%

From Tables 1-2, we can conclude that our DALES system provides reliable detection and counting of objects of interest in multi-camera environment, and that the multiple-object labelling is very accurate as well, outperforming state-of-the art techniques. DALES total precision to annotate multi-view videos is therefore very high.

For all the dataset, the average computational speed of our DALES software is in the range of few seconds, whereas the viewer function of DALES software takes only few milliseconds to process. Hence, our developed system could be used in context of online scene analysis.

4 Conclusions

Reliable, multi-view annotation of large amount of real-time visual data, such as surveillance videos or sport event broadcasts, is a challenging topic we have copped with. For this purpose, we have developed a new software tool called DALES which processes, in multi-camera environment, (i) multiple object-of-interest detection and (ii) counting, (iii) target segmentation and (iv) labelling, (v) image annotations, (vi) multi-stream video annotations and script generation. Moreover, our DALES software suits well as a viewer to display a loaded script with the text annotations of the multi-camera video sequence and the corresponding labelled multi-view images.

Our system shows excellent performance compared to the ones found in the literature, on one hand, for multiple-target detection and segmentation and, on the other hand, for object labeling. Multi-stream annotations with DALES are thus computationally efficient and accurate.

References

T. Alqaisi, D. Gledhill, and J. I. Olszewska. 2012. Embedded double matching of local descriptors for a fast automatic recognition of real-world objects. In *Proceedings of the IEEE International Conference on Image Processing (ICIP'12)*, pages 2385–2388.

A. Alsuqayhi and J. I. Olszewska. 2013. Efficient optical character recognition system for automatic soccer player's identification. In *Proceedings of the IAPR International Conference on Computer Analysis of Images and Patterns Workshop (CAIP'13)*, pages 139–150.

F. Archetti, C. Manfredotti, V. Messina, and D. Sorrenti. 2006. Foreground-to-ghost discrimination in single-difference pre-processing. In *Proceedings of the International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 23–30.

T. Athanasiadis, P. Mylonas, Y. Avrithis, and S. Kollias. 2007. Semantic image segmentation and object labeling. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(3):298–312.

L. Bai, S. Lao, G. J. F. Jones, and A. F. Smeaton. 2007. Video semantic content analysis based on ontology. In *Proceedings of the IEEE International Machine Vision and Image Processing Conference*, pages 117–124.

J. Black, T. Ellis, and P. Rosin. 2002. Multi View Image Surveillance and Tracking. In *Proceedings of the IEEE Workshop on Motion and Video Computing*, pages 169–174.

C++. 2011. C++ builder. Available online at: `https://downloads.embarcadero.com/free/c_builder`.

J.-W. Choi and J.-H. Yoo. 2013. Real-time multi-person tracking in fixed surveillance camera environment. In *Proceedings of the IEEE International Conference on Consumer Electronics*.

X. Dai and S. Payandeh. 2013. Geometry-based object association and consistent labeling in multi-camera surveillance. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 3(2):175–184.

R. Diaz, S. Hallman, and C. C. Fowlkes. 2013. Detecting dynamic objects with multi-view background subtraction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 273–280.

M. Evans, C. J. Osborne, and J. Ferryman. 2013. Multicamera object detection and tracking with object size estimation. In *Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 177–182.

N. Friedman and S. Russell. 1997. Image segmentation in video sequences: A probabilistic approach. In *Proceedings of the 13th Conference on Uncertainty in AI*.

S. Guler, J. M. Griffith, and I. A. Pushee. 2003. Tracking and handoff between multiple perspective camera views. In *Proceedings of the 32nd IEEE Workshop on Applied Imaginary Pattern Recognition*, pages 275–281.

R. M. Haralick. 1988. Mathematical morphology and computer vision. In *Proceedings of the IEEE Asilomar Conference on Signals, Systems and Computers*, volume 1, pages 468–479.

I. Haritaoglu, D. Harwood, and L. Davis. 2000. Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 77(8):809–830.

H.-H. Hsu, W.-M. Yang, and T. K. Shih. 2013. Multicamera object detection and tracking with object size estimation. In *Proceedings of the IEEE Conference Anthology*, pages 1–4.

W. Huang, Z. Liu, and W. Pan. 2007. The precise recognition of moving object in complex background. In *Proceedings of 3rd IEEE International Conference on Natural Computation*, volume 2, pages 246–252.

M. Izadi and P. Saeedi. 2008. Robust region-based background subtraction and shadow removing using colour and gradient information. In *Proceedings of the 19th IEEE International Conference on Pattern Recognition*, pages 1–5.

V. Kettnaker and R. Zabih. 1999. Bayesian multi-camera surveillance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1–5.

K. S. Kumar, S. Prasad, P. K. Saroj, and R. C. Tripathi. 2010. Multiple cameras using real-time object tracking for surveillance and security system. In *Proceedings of the IEEE International Conference on Emerging Trends in Engineering and Technology*, pages 213–218.

L. Lamard, R. Chapuis, and J.-P. Boyer. 2013. CPHD Filter addressing occlusions with pedestrians and vehicles tracking. In *Proceedings of the IEEE International Intelligent Vehicles Symposium*, pages 1125–1130.

P. Natarajan and R. Nevatia. 2005. EDF: A framework for semantic annotation of video. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, page 1876.

J. I. Olszewska and T. L. McCluskey. 2011. Ontology-coupled active contours for dynamic video scene understanding. In *Proceedings of the IEEE International Conference on Intelligent Engineering Systems*, pages 369–374.

J. I. Olszewska. 2011. Spatio-temporal visual ontology. In *Proceedings of the 1st EPSRC Workshop on Vision and Language (VL'2011)*.

J. I. Olszewska. 2012a. A new approach for automatic object labeling. In *Proceedings of the 2nd EPSRC Workshop on Vision and Language (VL'2012)*.

J. I. Olszewska. 2012b. Multi-target parametric active contours to support ontological domain representation. In *Proceedings of the RFIA Conference*, pages 779–784.

PETS. 2001. PETS Dataset. Available online at: `ftp://ftp.pets.rdg.ac.uk/pub/PETS2001`.

C. Stauffer and W. Grimson. 1999. Adaptive background mixture model for real-time tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

C. Town. 2004. Ontology-driven Bayesian networks for dynamic scene understanding. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, page 116.

K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. 1995. Wallflower: Principles and practice of background maintenance. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 1, pages 255–261.

B. Vrusias, D. Makris, J.-P. Renno, N. Newbold, K. Ahmad, and G. Jones. 2007. A framework for ontology enriched semantic annotation of CCTV video. In *Proceedings of the IEEE International Workshop on Image Analysis for Multimedia Interactive Services*, page 5.

C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland. 1997. Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785.

C. Yao, W. Li, and L. Gao. 2009. An efficient moving object detection algorithm using multi-mask. In *Proceedings of 6th IEEE International Conference on Fuzzy Systems and Knowledge Discovery*, volume 5, pages 354–358.

Z. Zivkovic and F. van der Heijden. 2004. Recursive unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(5):651–656.