

# FEATURE-BASED OBJECT MODELLING FOR VISUAL SURVEILLANCE

Gary Baugh, Anil Kokaram

Dept. of Electronic and Electrical Engineering,  
Trinity College Dublin, Ireland  
baugh@tcd.ie, anil.kokaram@tcd.ie

## ABSTRACT

This paper introduces a new feature-based technique for implicitly modelling objects in visual surveillance. Previous work has generally employed background subtraction and other image or motion based object segmentation schemes for the first step in identifying objects worthy of attention. Given that background subtraction is a notoriously noisy process, this paper investigates an alternative strategy by instead employing feature (SIFT [1]) clustering to characterise objects. The segmentation step is therefore performed on the sparse feature space instead of the image data itself. The paper also presents an application employing this idea for automatic detection of illegal dumping from CCTV footage. The Viterbi algorithm then allows robust tracking [2] of objects generated from the spatial clustering of these sparse foreground feature maps.

*Index Terms*— visual surveillance, SIFT, background modelling, foreground estimation

## 1. INTRODUCTION

Visual surveillance in dynamic scenes is currently a popular research topic in computer vision. The processing framework usually involves the following general stages: modelling the environment, object detection, classification and tracking of moving objects, and interpretation and description of behaviours. This paper addresses issues in the early stages of this process including environment modelling with object detection and tracking. The typical approach is first to segment objects of interest from the scene, and then to analyse the motion and appearance of those objects to identify suspicious or unusual activity. In this paper the problem is to detect when an illegal dumping event occurs at a *bring* centre or even in some protected area, i.e. when a person deposits trash in an area where it is not allowed.

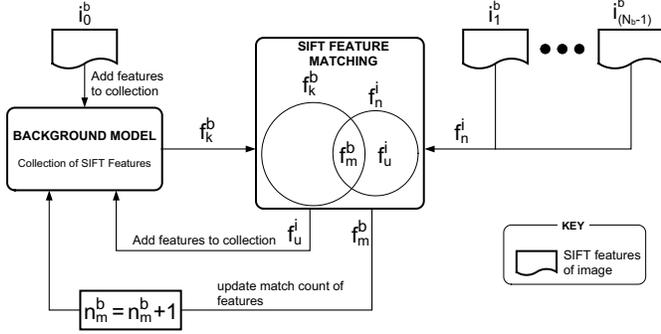
The problem of object segmentation in this scenario is differentiating between new objects in the scene and the natural behaviour of the environment which may include objects

leaving and entering the scene, or moving in some way. Background subtraction, when stationary cameras are used, is often used for this task. In that process, objects are identified in each frame by subtracting each current frame from a reference image of the background. Background subtraction is hard to use on its own because a stationary sensor does not necessarily imply a stationary background scene. Common examples of non-stationary background motion that abound in the real world are shadows, periodic motions, such as ceiling fans, clocks, and dynamic textures, such as fountains, swaying trees, or ocean ripples. One popular solution is to characterise the background using Gaussian mixtures in a statistical framework instead of relying on a single example of the background scene. These ideas were developed since 1997 by Wren [3], Stauffer [4], and Elgammal [5]. Elgammal in addition proposed nonparametric estimation methods to address the spatial correlation in the background activity. Alternative strategies for finding new objects in the scene involve motion based segmentation of some kind e.g. [6, 7]. Most flow computation methods are computationally heavy and very sensitive to noise, and can not be applied to real time application without specialized hardware.

Given the prevalence of work in the area of object detection in which feature based analysis has proven the only reliable mechanism [8], we propose here to use collections of SIFT features for articulating the segmentation process. This is very different from the pixel based schemes proposed so far. Using features rather than image data itself allows quite a few advantages. These include partial but implicit resistance to illumination changes, ability to cope with unusual motion activity, camouflaged foreground object detection, higher tolerance to background motion, and lower computation. In addition, the subsequent step in any surveillance system, that of tracking, is already facilitated by using features at the outset since the foreground objects are described by these feature vectors implicitly. This work uses the Viterbi strategy of Pitie et al [2] to track large numbers of objects in the scene based on SIFT descriptors.

---

This work was supported in part by SFI, Enterprise Ireland and Adobe Ltd



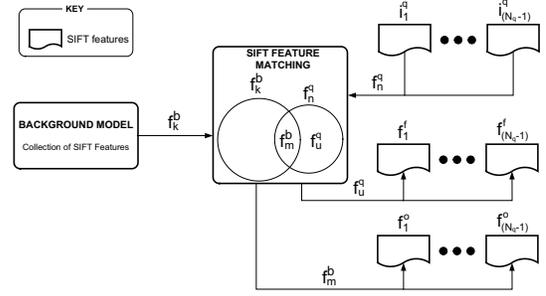
**Fig. 1.** The background model SIFT feature collection process finds the most significant and unique features from a sequence of background images  $i_n^b$ ,  $n = 0, 1, \dots, N_b - 1$ , to represent the background scene. The features of the first image  $i_0^b$ , constitute the initial set of features in the background model. The SIFT features for an image  $i_n^b$ , in the sequence are  $f_n^i$ . The features of the image  $f_n^i$ , are compared to the features in the background model  $f_k^b$ , to find the unmatched image features  $f_u^i$ , and the matched background features  $f_m^b$ . The match count,  $n_m^b$  incremented for the matched feature  $f_m^b$ , gives a quantitative measure of stability of the feature.

## 2. BACKGROUND MODELLING FRAMEWORK

The background is characterised by a set of SIFT feature locations and their associated descriptors. The  $k$ th feature in this set is defined as  $f_k^b$ . During the training process (shown in Fig. 1), the natural variation in the background is accounted for by accumulating new features depending on some measure of their persistence over the background training images  $i_n^b$ . Features are therefore matched between each new example image and the current reference set using their descriptors. The SIFT feature matching criterion proposed in [1] is used.

A persistence factor  $p_k^b$ , is defined for the  $k$ th background feature  $f_k^b$  as  $n_k^b/N_b$ . Here  $n_k^b$  is the number of training images over which that feature is matched, and  $N_b$  is the total number of example images. The set of features is then divided into a high persistence and a low persistence class by using a threshold  $p_t$ , typically  $p_t = 0.2$ . Define the low persistence group as  $f_k^l$ ,  $k = 1, 2, \dots, N_l$ , ( $p_k^b < p_t$ ) and the high persistence group  $f_k^h$ ,  $k = 1, 2, \dots, N_h$ , ( $p_k^b \geq p_t$ ). The idea behind doing this is to acknowledge that in the subsequent segmentation step, there is a heightened uncertainty in classifying a pixel  $(i, j)$  as foreground or background if it is in close proximity to a low persistent feature  $f_k^l$ . Regions of background with substantially large low persistence feature densities, must be completely ignored, or treated with less confidence when estimating foreground regions.

The amount of suppression required for these regions depends on the density of low persistence features. A suppression weight  $s_{i,j}^p$  is assigned to pixel  $(i, j)$  depending on the proximity of that site to the low persistence features  $f_k^l$ , with coordinates  $(i_k^l, j_k^l)$  in the entire image. The weight is calcu-



**Fig. 2.** The background model is a collection of SIFT features  $f_k^b$ . The foreground features  $f_i^f$ , of a query image  $i_i^q$ , are found by matching features of the query image  $f_n^q$ , with the background features  $f_k^b$ . The foreground features are the unmatched features  $f_u^q$ , of  $f_n^q$ . Matched background is estimated using the features  $f_i^o$ , which are the matched background features  $f_m^b$ , for the query image  $i_i^q$ .

lated as follows.

$$s_{i,j}^p = 1 - \frac{v_{i,j}^p}{\max_{i,j} (v_{i,j}^p)} \quad (1)$$

$$\text{where } v_{i,j}^p = \sum_{n=1}^{N_l} \exp \left( -\frac{(i - i_n^l)^2 + (j - j_n^l)^2}{2\sigma_s^2} \right)$$

$s_{i,j}^p$  is therefore a real number between 1.0 and 0.0.  $s_{i,j} = 0.0$  implies a site can not be considered foreground, while  $s_{i,j} = 1.0$  implies that it can. The variance  $\sigma_s$  depends on the spatial resolution of the image sequence.

## 3. FOREGROUND ESTIMATION

After background modelling, a rough segmentation of the example image into foreground and background helps to identify objects for subsequent tracking and analysis. Roughly speaking a foreground region is identified as a closely knit cluster of unmatched features from the example image. Figure 2 gives the process flow. Only a subset of foreground pixels are identified by the foreground features  $f_i^f$ . To gain a more complete foreground map, it is required to propagate the belief that a pixel is foreground to neighbouring pixels. Pixels in close proximity to a foreground or background feature point should have a heightened probability of being a member of a foreground or background region respectively. When a foreground object is introduced into a spatial region of a background scene, background features corresponding to this region are occluded. Hence, this region is populated with unmatched background features. This is quantified by an occlusion factor  $o_{i,j}^p$  as below in equation (2).

$$o_{i,j}^p = 1 - \frac{z_{i,j}^p}{\max (z_{i,j}^p)} \quad (2)$$

$$\text{where } z_{i,j}^p = \sum_{n=1}^{N_o} \exp \left( -\frac{(i - i_n^o)^2 + (j - j_n^o)^2}{2\sigma_o^2} \right)$$

Where  $(i_n^o, j_n^o)$  are the coordinates of  $f_n^o$  (matched background features). New background features created from the continuously changing background scene, will be observed as foreground features. However, the rate of generation of these new background features is assumed to be gradual. As a result, they will be surrounded by existing background features, which when matched, suppress or negate the effect of the new background features.

Foreground confidence weights,  $\beta_{i,j}^f$ , then combine the suppression factors from background modelling,  $s_{i,j}^p$  (equation 1), foreground estimation,  $o_{i,j}^p$  (equation 2), and the set of foreground features,  $f_i^f, i = 1, 2, \dots, N_f$ , with coordinates  $(i_i^f, j_i^f)$  to estimate foreground regions, given by:

$$\beta_{i,j}^f = s_{i,j}^p o_{i,j}^p \sum_{n=1}^{N_f} \exp\left(-\frac{(i - i_n^f)^2 + (j - j_n^f)^2}{2\sigma_b^2}\right) \quad (3)$$

The top row of Fig 3 shows this idea. The dots and  $\times$ s are matched and foreground feature points respectively, superimposed on  $\beta_{i,j}^f$ .

#### 4. TRACKING FOREGROUND REGIONS

The foreground confidence map  $\beta_{i,j}^f$  (3), contains foreground and background regions. Foreground candidate regions are identified as  $\beta_{i,j}^f \geq b_{min}^t$ , where  $b_{min}^t$ , is the mean of all the non-zero weights in  $\beta_{i,j}^f$ . This estimated foreground binary mask is defined as  $\zeta_{i,j}$ . Each object blob in this mask is labelled uniquely,  $b_n$  is the  $n$ th blob. (We use the watershed algorithm here for convenience). The weights of the blob  $b_n$  in the foreground confidence map  $\beta_{i,j}^f$  are  $\omega_{i,j}^n = \beta_{i,j}^f \zeta_{i,j}^n$ , where  $\zeta_{i,j}^n$  is the region corresponding to the  $n$ th blob.

##### 4.1. Blob Description and Classification

After blob labelling, a vector descriptor  $\vec{b}_n$  is formed for each blob  $b_n$ . The entries in  $\vec{b}_n$  are: The area  $A_n$ , weighted spatial location  $(x_n, y_n) = \frac{1}{\sum_{i,j} \omega_{i,j}^n} \left( \sum_{i,j} i \omega_{i,j}^n, \sum_{i,j} j \omega_{i,j}^n \right)$ , mean foreground belief weight  $\mu_n = \frac{1}{A_n} \sum_{i,j} (\omega_{i,j}^n)$ , maximum radius  $r_n$  from  $(x_n, y_n)$ , blob weight  $\eta_n = A_n u_n^2$ , and colour  $\vec{c}_n = \langle \vec{r}_n, \vec{g}_n, \vec{b}_n \rangle$ , where  $\vec{r}_n, \vec{g}_n$ , and  $\vec{b}_n$  are the normalized bin values (32 bins) of the histogram for the 8-bit RGB image segment corresponding to the blob.

The effects of the suppression weights  $s_{i,j}^p$ , and the occlusion weights  $o_{i,j}^p$  are to generally make foreground blob weights  $\eta_n^f$ , significantly larger than background blob weights  $\eta_n^b$ .

##### 4.2. Temporal Blob Filtering

At this stage, there is a set of blobs with vector descriptors that represent foreground and background regions. The blobs

in the set with significantly small weights are discarded since they are assumed to represent background regions. In a real world application, there are image sequences that contain no foreground objects. Since there are no foreground blobs with large weights to aid in background blob discrimination, background blobs are inevitably selected as foreground. However, from observation of real world data, these background blobs do not persist spatio-temporally. They appear for a relatively brief period time in a video sequence. On the other hand, foreground blobs have smooth transitions in time and space throughout a video sequence, and are persistent, allowing them to be tracked using the Viterbi strategy in [2]. Tracking blobs in temporal adjacent example images produces sequences of blobs. The blob sequences of legitimate foreground objects are longer than those of background blobs. Only blob sequences of a minimum length (50 frames for this application) are considered to be generated by foreground objects, and referred to as legitimate. Accordingly, valid foreground blobs must correspond to a legitimate blob sequences.

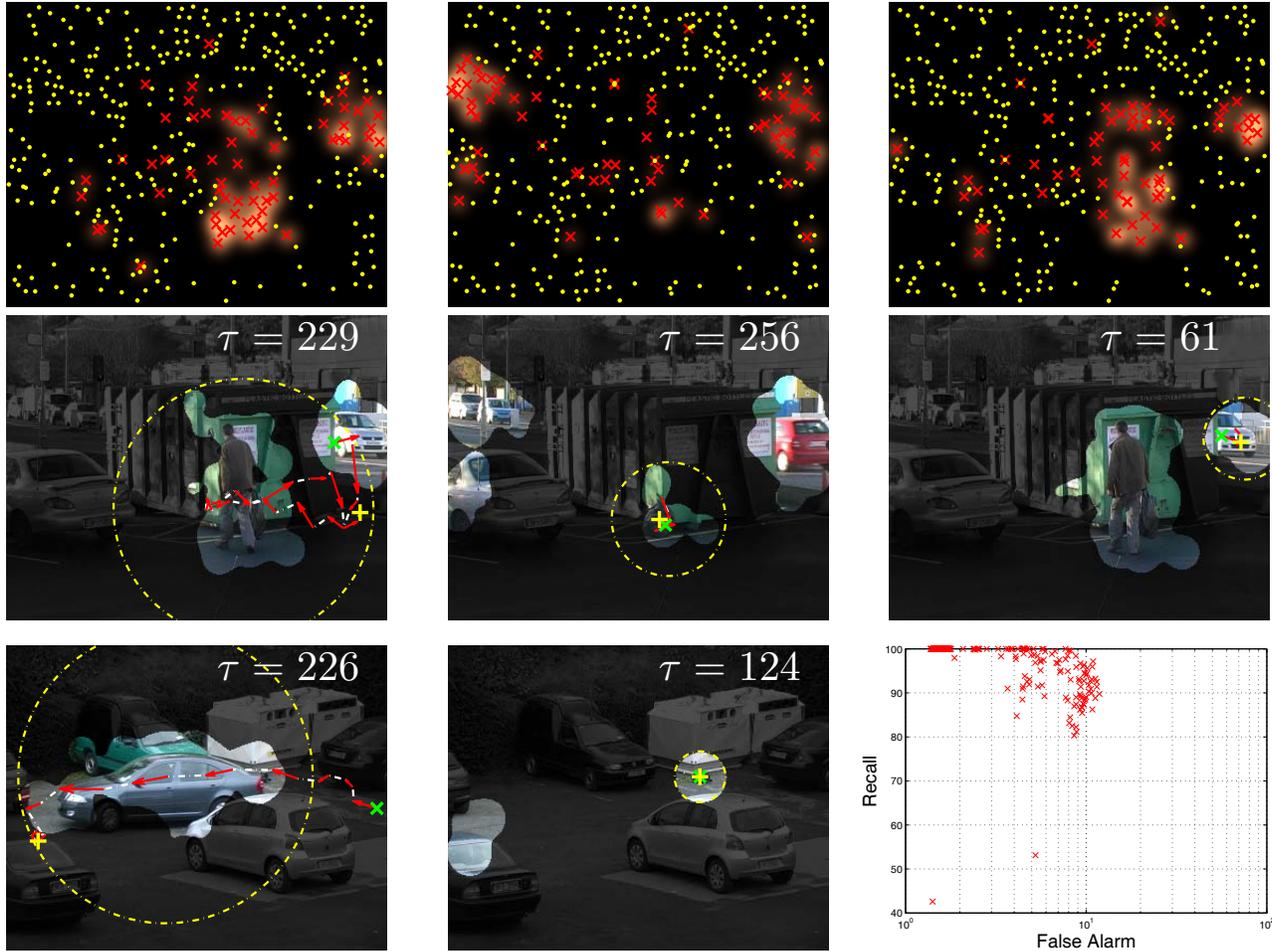
## 5. RESULTS

The same parameters were used for all test videos ( $N_b = 150, \sigma_b = 6, \sigma_s = 15, \sigma_o = 15$ ). The bottom row of figure 3 demonstrates the effectiveness of the proposed algorithm in tracking foreground objects that have been partially occluded. Even the relatively small bag that was thrown from the moving car, was successfully detected and tracked throughout the video sequence (middle image). The average false alarm and recall rates for this test video are 2.6% and 98.8% respectively. There was moderate background motion due to swaying leaves. Generally, the algorithm is quite capable of handling outdoor scenes of this nature.

The middle row of figure 3 demonstrates a more challenging outdoor scene, that has large background movement due to vehicular traffic. Although legitimate foreground objects were detected and tracked for this test sequence (left and middle images), some background objects were classified as foreground (right image). The merging of foreground and background blobs distort the motion paths of the foreground objects.

## 6. CONCLUSION

We have presented a novel scheme for object characterisation using collections of features instead of background subtraction. The new scheme holds great potential for robust object identification for video surveillance. The main focus of the work presented here was object detection, and due to the nature of the application investigated, imperfect segmentation was tolerable. However, further work will involve the probabilistic modelling of the framework in order to improve the connection between the sparse feature maps and the corresponding foreground region based segmentation.



**Fig. 3.** Top row: Foreground ( $\times$ ) and matched ( $\cdot$ ) feature points superimposed on  $\beta_{i,j}^f$ , for the example images in the middle row. Middle row: Blob sequences for a 687 frames test video. The arrows indicate the transition of the centers of the blobs for a blob sequence which is  $\tau$  frames long. The blobs shown (circled) occur in the middle of the sequences, where the starts and ends of the sequences are indicated with  $\times$ s and  $+$ s respectively. Bottom row: Blob sequences for a 1099 frames test video (left and middle images), right image: Plot of the recall/false alarm rates (in percentages) calculated per frame for this video, using 641 frames of ground truth data.

## References

- [1] D. Lowe, "Distinctive image features from scale-invariant keypoints," in *International Journal of Computer Vision*, 2003, vol. 20, pp. 91–110.
- [2] F. Pitie, S-A. Berrani, R. Dahyot, and A. Kokaram, "Off-line multiple object tracking using candidate selection and the viterbi algorithm," in *IEEE International Conference on Image Processing*, Genoa, September 2005.
- [3] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780–785, 1997.
- [4] C. Stauffer, W. Eric, and L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 747–757, 2000.
- [5] A. Elgammal, R. Duraiswami, D. Harwood, and L.S. Davis, "Background and foreground modeling using nonparametric kernel density for visual surveillance," in *Proceedings of the IEEE*, July 2002, vol. 90, pp. 1151–1163.
- [6] A.J. Lipton, "Local application of optic flow to analyse rigid versus non-rigid motion," in *ICCV Workshop on Frame-Rate Vision*, Sep. 1999.
- [7] D. Meyer, J. Denzler, and H. Niemann, "Model based extraction of articulated objects in image sequences for gait analysis," *Proc. IEEE Int. Conf. Image Processing*, vol. C, pp. 78–81, 1998.
- [8] B. Schiele, "Model-free tracking of cars and people based on color regions," *Image and Vision Computing*, vol. 24, no. 11, pp. 1172–1178, November 2006.