Neuromorphic Bayesian Surprise for Far-Range Event Detection

Randolph C. Voorhies, Lior Elazary and Laurent Itti

Department of Computer Science, University of Southern California

Abstract

In this paper we address the problem of detecting small, rare events in very high resolution, far-field video streams. Rather than learning color distributions for individual pixels, our method utilizes a uniquely structured network of Bayesian learning units which compute a combined measure of "surprise" across multiple spatial and temporal scales on various visual features. The features used, as well as the learning rules for these units are derived from recent work in computational neuroscience. We test the system extensively on both real and virtual data, and show that it outperforms a standard foreground/background segmentation approach as well as a standard visual saliency algorithm.

Figure 1. Example frame from desert test data with inset showing two pedestrian targets

1. Introduction

When processing video under real-time constraints, detecting salient events plays a critical role by selecting subsets of the scene towards which limited computational resources should be allocated. Event detection becomes increasingly difficult as the field of view increases, and the number of pixels-on-target decreases. For example, figure 1 shows a scene in which two pedestrians comprising roughly 15 vertical pixels are traversing a desert scene captured by a 16 mega-pixel video camera. These situations are commonly encountered in border surveillance applications, yet to our knowledge no system exists which can robustly detect anomalous events in such circumstances. In this paper, we focus on developing a solution to this difficult problem.

Current research in video surveillance is concerned mainly with the detection and annotation of high-level events [16, 1, 9, 15]. While they often provide very rich semantic descriptions of the scene, these approaches require a significant number of pixels-on-target, and are thus unsuitable to our application. More traditional low-level background subtraction algorithms [12, 14, 4, 17, 10] generally operate on pixel level statistics and so are more applicable to our constraints. These methods typically operate by maintaining a model of the background, e.g. as a mixture of Gaussian distributions [12], which is learned online to reflect the state of the world. However, as these methods have generally served as input into higher level tracking and recognition algorithms, they perform best when segmenting large regions of foreground from background. Thus little attention has been paid towards selecting representations which provide an acceptable signal to noise ratio for far range surveillance.

This paper introduces a novel method for the detection of such small, rare events in far-field surveillance footage. We leverage recent computational neuroscience research by [6] to build a network of Bayesian learners which is inspired by models of the primate visual attention system. These Bayesian learners adapt to changes in a scene by learning feature statistics over several spatial and temporal scales. By comparing each unit's learned prior probability distribution before a video frame is presented to the posterior distribution estimated after the frame, we generate a measure of "surprise" which we show is a robust measure for detecting the presence of unexpected events. This is because

our learned background distributions encode both spatial and temporal information, such as waving tree branches or shimmering atmospheric distortions. We test our system on scenes staged in a desert environment in which transient targets (persons, vehicles) appear and disappear several kilometers away from a high-resolution camera (16 megapixels) with a field of view of $\sim 20^\circ$, such that each target covers as few as ten pixels in the longest dimension. We also test our system on artificially generated data to more precisely characterize its efficiency in detecting when and where such targets appear.

In section 2, we detail the construction of our model and implementation. In section 3, we test the system on real and artificial targets, and compare the performance of our system to a standard foreground detection system typically used in surveillance applications. In section 4, we review the results of these tests, and in section 5 we provide our conclusions.

2. Methods

Rather than trying to determine which pixels in an image belong to foreground versus background regions, we pose our problem as determining when and where the spatial and temporal statistics of an image constitute a "surprising" event warranting the attention of a higher-level process or human operator. To do so, we first decompose the image into a number of low-level image features, as described in section 2.1. At each location in each feature map, the statistics of features are then learned online through a network of Bayesian computation units and combined, as described in section 2.2. We run our implementation in real-time by breaking the 16 megapixel input images into 256×270 pixel "chips," and running each chip in parallel. The system then emits the maximum surprise value computed within each chip, which can be thresholded to trigger event detections.

2.1. Feature Detectors

Several types of feature detectors were chosen for our implementation, all of which have been well documented in the literature on the primary visual cortex of mammals. Each detector type is implemented an approximation to center-surround cells with receptive fields spanning a range of sizes. This approximation is efficiently performed by constructing Gaussian image pyramids [3] spanning from 2 octaves to 8 octaves above the original scale. In total, we create 12 different low-level feature pyramids (one intensity, two color, four orientation, four motion, one flicker) which are combined to create a total of 72 center surround feature detector maps.

To create the center-surround channels, we first create their component feature pyramids as follows, following the basic approach already explored previously to develop saliency map algorithms [8]. From an input image comprised of a red, green, and blue channel (r, g, and b), we compute an intensity pyramid (I) as well as four color pyramids (R, G, B, Y) which are tuned to respond to red, green, blue, and yellow respectively while producing zero response to both black and white. Four pyramids are also created for orientations, denoted as O_θ and are made by convolving the input image with an oriented Gabor filter. Four motion energy pyramids (M_θ) are created as responses to Reichardt motion detectors [13] in the four cardinal directions. Finally, a flicker pyramid (F) is created by subtracting the intensity channel response at the current frame from the intensity channel response at the previous frame.

From these raw channel pyramids, the various types of center-surround feature detectors are computed by subtracting the value of a center channel at level $x = \{2, 3, 4\}$ in the pyramid by the response of a rescaled and interpolated surround channel at level $x + \delta$, with $\delta = \{3, 4\}$, giving us six center/surround pairs: $2/5$, $2/6$, $3/6$, $3/7$, $4/7$, and $4/8$.

Each of these feature maps is sensitive to local contrasts at multiple spatial scales in its feature domain. $\mathcal{I}(c, s)$ is an intensity contrast map, $\mathcal{RG}(c, s)$ and $\mathcal{BY}(c, s)$ are red/green and blue/yellow double opponency maps, $\mathcal{O}_\theta(c, s)$ is an orientation contrast map, $\mathcal{M}_\theta(c, s)$ is a motion contrast map, and $\mathcal{F}(c, s)$ is a flicker contrast map. Using a total of six center/surround combinations as described above and four orientations yields the total of 72 center-surround feature maps.

2.2. Surprise computation

In [5, 7], surprise is defined in general Bayesian terms, as the difference between a prior belief distribution and a posterior distribution that is generated when new data is observed. Assuming a prior distribution $P(M)$ of beliefs over a set of hypotheses or models $\mathcal{M}$, Bayes rule is employed to compute the posterior beliefs $P(M|D)$ each time new data D is observed:

$$P(M|D) = P(M)P(D|M)/P(D) \tag{1}$$

Intuitively, if the posterior distribution is highly similar to the prior, then the data D contributed little change to the observer's beliefs, and hence it was boring; conversely, data that yields a significant update of the prior into the posterior is surprising as it significantly affected the learner's beliefs. This prompted a simple definition of surprise $S(D, \mathcal{M})$ from first principles in [5], as some distance measure between the posterior and the prior; here we use the Kullback-Leibler divergence KL, such that:

$$S(D, \mathcal{M}) = KL(P(M|D), P(M)) \tag{2}$$

$$= \int_{\mathcal{M}} P(M|D) \log \frac{P(M|D)}{P(M)} dM \tag{3}$$

We implement our system using such surprise as the basic measure of how interesting video data may be over time. A single surprise computation unit thus takes as input a prior distribution and a data distribution, and outputs both a posterior distribution as well as a surprise value representing the amount of novelty in the input.

In our implementation, a chain of such surprise detection units is connected to each location in each level of each feature map such that the first unit in the chain estimates its prior distribution directly from a feature detector's output value, as further described below. Data is obtained from each video frame, and is modeled at every visual location as a Poisson distribution $M(\lambda)$, where the rate parameter λ is simply estimated as the detector's output value. Modeling the prior and posterior as Gamma distributions:

$$P(M(\lambda)) = \gamma(\lambda; \alpha, \beta) = \frac{\beta^{\alpha} \lambda^{\alpha-1} e^{-\beta\lambda}}{\Gamma(\alpha)} \quad (4)$$

with shape $\alpha > 0$ and inverse scale $\beta > 0$ parameters and $\Gamma(.)$ the Euler Gamma function, allows us to use the posterior from a previous step as the prior for the current step in a Bayesian learning configuration. Given an observation $D = \overline{\lambda}$ and prior distribution $\gamma(\lambda; \alpha, \beta)$, the posterior $\gamma(\lambda; \alpha', \beta')$ obtained by Bayes' rule is also a Gamma distribution, with:

$$\alpha' = \alpha + \overline{\lambda} \quad \text{and} \quad \beta' = \beta + 1 \quad (5)$$

and the surprise that results from observing D can be computed in closed form [7]:

$$S(D, \mathcal{M}) = KL(\gamma(\lambda; \alpha', \beta'), \gamma(\lambda; \alpha, \beta)) = \quad (6)$$

$$\alpha \log \frac{\beta'}{\beta} + \log \frac{\Gamma(\alpha)}{\Gamma(\alpha')} + \beta \frac{\alpha'}{\beta'} + (\alpha' - \alpha)\Psi(\alpha') \quad (7)$$

where $\Psi(.)$ is the digamma function. In this way, the detector is able to constantly update its belief of the distribution of a single feature detector's response and to output a measure of the amount by which a new feature response violates its learned expectations.

By applying such a surprise detector to each feature map at each spatial scale, we are able to detect unexpected events of various sizes and types. However, to effectively adjust to repeating or periodic motions in the image (e.g., trees in the wind), it is also desirable to detect surprise at multiple temporal scales. In addition, while monitoring surprise over time may be useful to detect local transient events, computing surprise over space is also important to focus onto those events which are also spatially salient (see [7] for further details). To accomplish this, we set up a network of these surprise detectors to form a complete temporal and spatial surprise computation unit. The input to this unit is a single feature map from a single pyramid scale. At each location in this map, we compute both the spatial and temporal surprise over five temporal scales. At the first temporal scale, the data from the feature map is fed into the data inputs of both a temporal surprise detector, and a spatial surprise detector. The temporal surprise detectors at each temporal scale use the posterior from their previous time step as the prior, and output the amount of surprise generated. The posterior from each temporal detector is then used as the data input for the next-level temporal detector. In this way, each successive level of the temporal surprise computation chain adapts to a slower band of temporal frequencies than its input. The spatial surprise detectors use the same data input as the temporal detectors, but use an average of the surrounding responses weighted by a Gaussian envelope as their prior distributions. This spatial surprise detector then outputs the novelty of a stimulus based only on the surrounding feature responses from the current frame. At each temporal scale, the spatial and temporal surprise computations are combined with a weighted sum and a non-linearity. The resulting maps from each temporal scale are then combined via a product to produce the final surprise map for the given feature. Once all feature maps are computed, they are averaged to produce a final surprise map for the current frame.

3. Performance Evaluation Methods

To analyze the effectiveness of our attentional front-end in highlighting only regions which contain interesting events, we put our system through two sets of rigorous tests and compare our results to those obtained by a standard foreground/background discrimination algorithm as well as a neuromorphic spatial attention model.

3.1. Comparison Algorithms

The foreground/background discrimination algorithm used as benchmark is the popular OpenCV [2] implementation of Li et al's 2003 paper, "Foreground Object Detection from Videos Containing Complex Background" [10], which we believe is a good example of the canonical pixel-based background learning methods. Li's algorithm uses a Bayesian decision rule and feature vector learning to produce a dense map denoting the probability that each pixel belongs to a foreground object rather than the background. The maximum pixel's likelihood of containing a foreground object within each region was then taken as the score for that region and these scores were then aggregated together as in our implementation.

The spatial attention model used for comparison is an integer-math implementation of Itti's saliency model [8, 11]. This model uses the same feature detector types, pyramid architecture, center-surround computations, and ranges of spatial scales as our proposed algorithm, but instead of

computing surprise, applies a non-linear spatial competition within each feature map and across all scales. Using this model as reference allows us to quantify how much of the performance of our new method is attributable to the surprise component *per se*, as opposed to the choice of low-level visual features and range of pyramid scales used. The feature maps are then summed together to produce a master saliency map. As with Li's algorithm, this saliency model was applied to our distributed architecture such that each image region was computed independently in parallel.

3.2. Testing and validation experiments

Two sets of tests were run using all of the algorithms described in section 3.1 to determine their performance in both the real world, and in rigorously generated artificial scenes. The first set of tests was used to establish the usefulness of each algorithm in a realistic large scale surveillance scenario, while the second set of tests explores the performance of each algorithm by varying the difficulty of the detection task using target sizes and image noise as parameters.

In the first set of tests we were provided with three ground-truthed datasets of video taken from a desert environment. Each dataset consisted of 4864×3248 pixels shot at 2.98 frames per second, with each frame comprising a 20° horizontal field of view. There were a total of 5333 frames from these videos, totalling 29.8 continuous minutes of footage. The videos were hand annotated to provide ground truth bounding boxes around targets of interest. Throughout all video sequences, these annotated targets had an average size of 20.1 ± 17.5 pixels by 37.1 ± 34.9 pixels and ranged from around .5km to nearly 10km away from the camera.

In the second set of tests, we created a large set of scenes with artificial targets in which both the size of the targets and the level of noise in each clip was independently adjustable. These artificial target events were created by extracting 200 consecutive frames of size 256×270 pixels from each of 20 different regions from the original datasets which contained no events, documented or otherwise. Five different images of targets were then composited across the scenes at three different scales onto these backgrounds to create a series of 300 video clips. The spatial scales were defined as the number of pixels across the largest dimension of the target, and the targets were presented at scales of 10, 40, and 80 pixels. These clips were then corrupted by five different levels of Gaussian noise.

The evaluation of both test scenarios was achieved by creating a log file containing the maximum score for each region in each frame. In the second round of tests, each frame contained only one region, but the analysis remained the same. These log files were then compared against the respective ground truths to create a Receiver Operating Characteristic (ROC) plot of the probability of a true positive versus the probability of a false positive, for a surprise threshold which was swept from the minimum observed region score to the maximum. A given region in a frame was recorded as containing a true positive for a given threshold value if its maximal surprise value exceeded that threshold and contained at least 25% of the width and height of the bounding rectangle of a ground-truthed target. A chip was recorded as a false positive if it exceeded the threshold, but overlapped with less than 25% of the width and height of the bounding rectangle of a ground-truthed target. The probability of a true positive for a given threshold was then calculated as the total number of true positives divided by the number of regions which actually contained targets. The corresponding probability of a false alarm for that threshold was calculated as the total number of false alarms divided by the number of regions which did not contain targets.

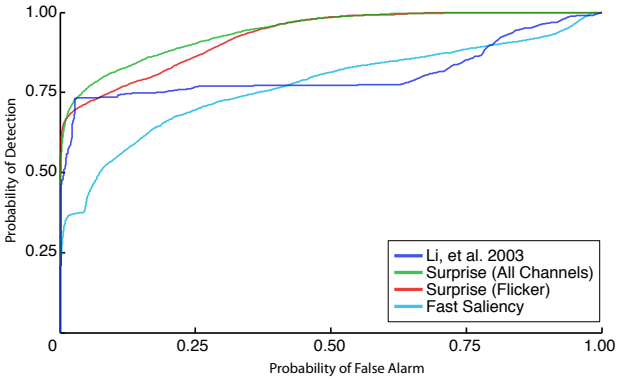

Figure 2. ROC results from Desert Environment Tests

4. Results

The ROC results from the first round of testing, in which real events were staged in desert scenes, are shown in figure 2. The plot shows the probability of the system reporting a false positive along the horizontal axis, and the probability of the system reporting a true positive along the vertical axis. The two versions of the surprise algorithm which were run using different sets of feature detectors are labeled as 'Surprise (All Channels)' and 'Surprise (Flicker).' The 'Surprise (All Channels)' model utilized feature detectors for color, flicker, intensity, orientation, and motion, while the 'Surprise (Flicker)' model used only a flicker feature detector. The OpenCV implementation of Li's algorithm is labeled in the plot as 'Li, et al. 2003', and the integer implementation of Itti's saliency model is labeled as 'Fast Saliency.' The area under each of these curves is shown in table 1.

The results of the second round of testing in which artificial target events were created are shown in Fig. 3. In these tests, the two surprise models had a much stronger invari-

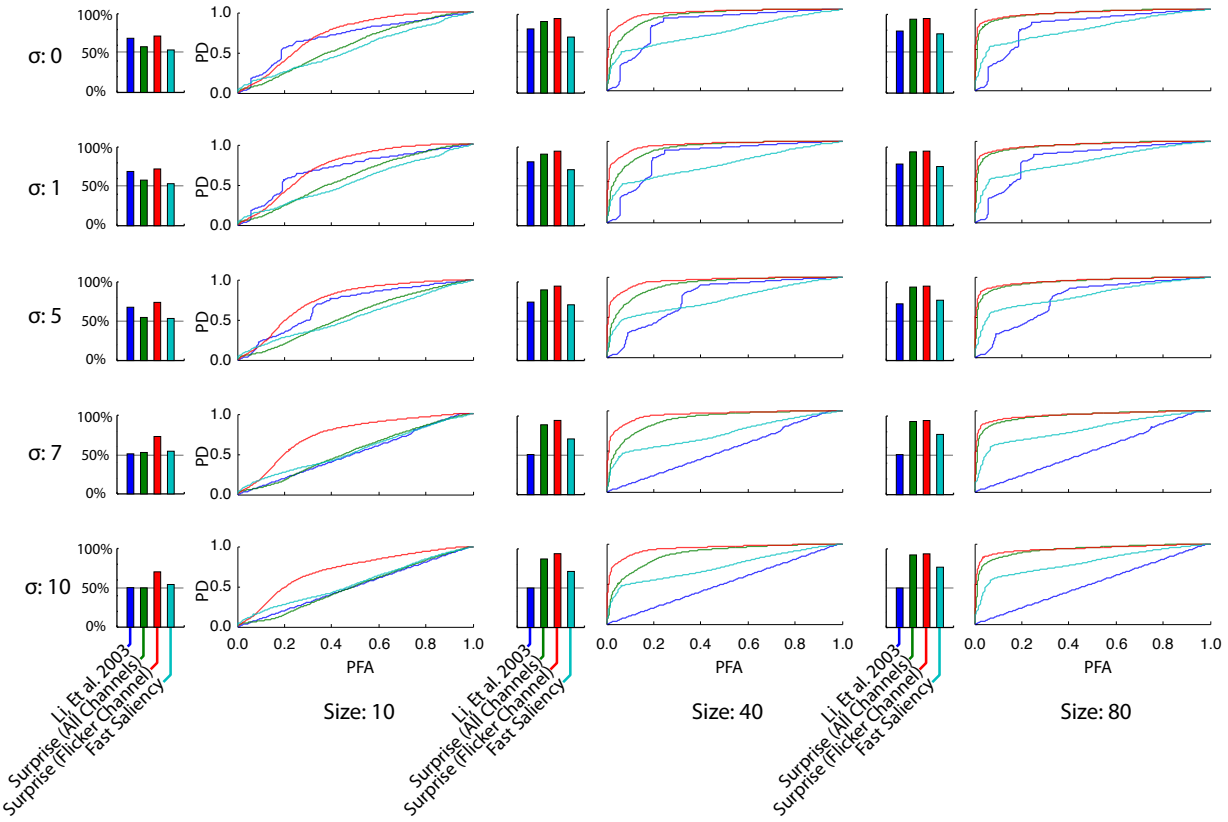

Figure 3. Test results for artificial target video datasets generated with Gaussian noise shown as Receiver Operating Characteristic (ROC) plots with the probability of a false positive (PFA) along the X-axis, and the probability of detection (PD) shown along the Y-axis. Increasing levels of noise are shown down the rows of each figure, and increasing target sizes are shown across the columns. The areas under each ROC curve are summarized in bar graphs to the left of each ROC plot.

Algorithm	AUC Score
Li, et al. 2003	0.80754
Surprise (All Channels)	0.93957
Surprise (Flicker)	0.92502
Fast Saliency	0.77038

Table 1. AUC Scores for Tests on Real Data

ance to the amount of noise than the reference algorithms. Additionally, while all of the systems had difficulty detecting the extremely small 10 pixel targets, the surprise model with all feature detectors enabled showed poor relative performance in the low noise case. Using only the flicker detector allowed the model to achieve the best performance in all cases. A random permutation test with 10,000 iterations showed that the flicker surprise model outperformed the other systems with a better than 5% significance level in all cases. Comparing the ROC curves for surprise with all channels vs. flicker only reveals that, for a given hit rate, the false-alarm rate with all channels is higher than when using only flicker, revealing that the surprise model with all channels does not score as well overall mainly because it detects more false events. That is, the flicker channel has higher signal-to-noise ratio, in our artificial dataset, for targets vs. non-targets than does the combination of all channels. In the following section we discuss this result in light of the results obtained with the first experiment.

5. Discussion

In this work, we have presented a novel application of a neuroscience attention model to the field of large-field-of-view surveillance. A first set of tests showed that the system was able to reliably detect subtle targets in a real environment, which more often evaded detection by a standard foreground segmentation algorithm as well as a popular spatial visual attention model. The second testing paradigm revealed the system's strong invariance to both noise and target size. Even under noise conditions which made the other algorithms perform only slightly above chance, the surprise

model with flicker feature detectors took only a modest performance hit.

While the model performed best in the first task with all of its feature detectors enabled, it was interesting to note that it performed best in the second task with only the flicker channel enabled, as can be seen in figure 3. The flicker channel produced the best results in these tests because the artificial targets had been designed by hand to have similar hues and contrasts ratios to the background, in order to increase the test difficulty. This similarity diminished the effectiveness of the color, intensity, and orientation channels because they rely highly on static contrast differences between foreground and background. In essence, in the second test, channels other than flicker hence contributed little in terms of increasing hit rate over the flicker channel alone, but added to the false alarm rate; in sum, using all channels resulted in lower ROC performance compared to flicker alone. However, as can be seen in figure 2, the full set of features provided a better probability of detection for low false alarm rates in the real world scenarios. This is because real world targets are likely to have spatial differences against backgrounds, as well as because real targets may generate little detectable motion. This suggests that while our test sets may have been biased slightly towards the flicker channel responses, real world applications will likely require the full range of feature detectors.

References

[1] E. Andrade, S. Blunsden, and R. Fisher. Hidden markov models for optical flow analysis in crowds. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 1, pages 460 –463, 0-0 2006.

[2] G. Bradski. *Open Source Computer Vision Library*. Prentice Hall, 2004.

[3] P. J. Burt and E. H. Adelson. The laplacian pyramid as a compact image code. *IEEE Trans on Communications*, 31:532–540, 1983.

[4] A. Elgammal, D. Harwood, and L. Davis. Nonparametric model for background subtraction. In D. Vernon, editor, *Computer Vision ECCV 2000*, volume 1843 of *Lecture Notes in Computer Science*, pages 751–767. Springer Berlin / Heidelberg, 2000.

[5] L. Itti and P. F. Baldi. A principled approach to detecting surprising events in video. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 631–637, San Siego, CA, Jun 2005.

[6] L. Itti and P. F. Baldi. Bayesian surprise attracts human attention. In *Advances in Neural Information Processing Systems, Vol. 19 (NIPS*2005)*, pages 547–554, Cambridge, MA, 2006. MIT Press.

[7] L. Itti and P. F. Baldi. Bayesian surprise attracts human attention. *Vision Research*, 49(10):1295–1306, May 2009.

[8] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, Nov 1998.

[9] M. W. Lee and R. Nevatia. Human pose tracking in monocular sequence using multilevel structured models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(1):27 –38, jan. 2009.

[10] L. Li, W. Huang, I. Y. H. Gu, and Q. Tian. Foreground object detection from videos containing complex background. In *Proceedings of the eleventh ACM international conference on Multimedia*, MULTIMEDIA '03, pages 2–10, New York, NY, USA, 2003. ACM.

[11] R. J. Peters and L. Itti. Applying computational tools to predict gaze direction in interactive visual environments. *ACM Transactions on Applied Perception*, 5(2):Article 8, 2008.

[12] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2, pages 2 vol. (xxiii+637+663), 1999.

[13] V. Torre and T. Poggio. An application: a synaptic mechanism possibly underlying motion detection. In W. E. Reichardt and T. Poggio, editors, *Theoretical approaches in neurobiology*, pages 39–46. The M.I.T Press, Cambridge, MA, 1978.

[14] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: principles and practice of background maintenance. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 255–261 vol.1, 1999.

[15] B. Zhao, L. Fei-Fei, and E. P. Xing. Online detection of unusual events in videos via dynamic sparse coding. In *The Twenty-Fourth IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011.

[16] T. Zhao, R. Nevatia, and B. Wu. Segmentation and tracking of multiple humans in crowded environments. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(7):1198–1211, 2008.

[17] J. Zhong and S. Sclaroff. Segmenting foreground objects from a dynamic textured background via a robust kalman filter. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 44 –50 vol.1, oct. 2003.