

Detection of Unusual Motion Using Computer Vision

Cláudio R. Jung , Julio C. S. Jacques Jr., John Soldera and Soraia R. Musse
PIPCA - Graduate School on Applied Computing - Universidade do Vale do Rio dos Sinos
Av. UNISINOS, 950. São Leopoldo, RS, Brasil, 93022-000

Abstract

In this paper, we propose different criteria for detecting unusual motion in surveillance cameras. Initially, a certain environment is observed within a time interval, and captured trajectories are used as examples of usual trajectories. These trajectories are used to build a Spatial Occupancy Map (SpOM, which is introduced in this paper) of the observed people, as well as main flow directions. In the test period, each new trajectory is classified as normal or unusual with respect to spatial occupancy and trajectory consistency. The spatial occupancy criterion considers the relation of space occupancy between the new tracked trajectory and the observed period. The trajectory consistency criterion considers the agreement of the new trajectory with the main flows extracted in the training period. Experimental results showed that these criteria can be used as an automatic pre-screening of suspect motion in surveillance applications.

1 Introduction

A relevant problem in computer vision is the automatic detection of suspect motion in filmed video sequences. In fact, surveillance cameras are becoming a popular solution for crime prevention, and the development of automatic techniques to detect suspect motion could improve the efficiency of surveillance systems.

The definition of suspect motion is very context-dependent, and it is very difficult to define general criteria to identify suspect movements in a general context. A more treatable problem is the definition of unusual motion, given a training set (the observed period) of “normal” trajectories. We believe that one could qualify the motion of a given person as normal/unusual with respect to the following criteria:

1. Spatial occupancy: a new trajectory can be considered unusual if it comprises a spatial region significantly different than the training period.
2. Trajectory coherence: a person may walk in region that

was extensively occupied in the training period, but present an abnormal behavior with respect to trajectory consistency (e.g. all people in the training period move east-west, but the new tracked person moves in the same region in the opposite direction west-east).

3. Interaction with other people: in certain situations, only the trajectory itself is not enough to characterize suspect motions, as people interact among them. For instance, two persons may walk coherently with respect to the previous criteria, but one could be suspiciously following the other.

We also believe that several other criteria may be considered for specific applications, such as people carrying objects, moving with suspicious poses, etc. However, in this work we focus our analysis in the first two criteria: spatial occupancy and trajectory coherence.

The remainder of this paper is organized as follows. Section 2 presents an overview of techniques based on computer vision for detection unusual movements. The proposed algorithm is provided in Section 3, and experimental results are given in Section 4. Finally, our conclusion are drawn in Section 5.

2 Related Work

Several researchers have been analyzing the problem of detecting unusual motion in filmed video sequences. Next, some of these techniques are briefly revised (a more extensive review on visual surveillance systems can be found in [10]).

Boghossian and Velastin [1] discussed how to detect dangerous situations in large crowds and how to detect walls and strangle points on the way of such people. They focused on detecting such events using motion vectors and segmentation of the crowd flow. A large crowd moving in a busy space is classified as non-rigid elastic motion and the results of this system can be assisted in real-time by an operator to improve crowd safety.

Davies et al. [5] described a model using field theory and flow dynamics in order to discuss crowd behaviors, and

also to provide the expertise to suggest solutions for crowd monitoring and control based on image processing and understanding. The main theme of their research is to find general models of crowd behavior, which do not rely upon detecting the behavior of individuals.

Others works are focused on tracking people individually. Stauffer and Grimson [17] worked with statistical methods to track people and other objects, achieving robustness on day and on night. In their work, objects are classified and recognized through prior models. The system learns with such recognitions and accumulated prototype motion histogram from filmed recognized sequences, that allow the system to detect unusual events. An inspector watching may help to classify the level of danger of that scene. In this work, they are more concerned with people recognition and their interaction with objects for surveillance applications.

Buxton and Cong [2] describe their method to determine individual people behavior using Bayesian networks. First, a camera is calibrated with a three-dimensional model of the area, which is previously described. Objects are recognized through hierarchical segmentation of the roundabout ground-plane. The object's dynamic is tracked and its behavior is described as Bayesian networks, which contain information such as time and events. To classify events, the system detects the proximity of agents as not near, nearby, close, very close and touching.

The methods described in [12–14] use the concept of trajectory clustering to detect unusual patterns. In [13], trajectories are grouped according to a distance-based similarity metric in a training period, generating different clusters. Then, a Bayes network is used to determine in which cluster the new trajectories should be grouped. If no cluster is found, the analyzed trajectory is detected as unusual. In addition, a grammar is used in order to provide textual semantics for the tracked trajectories. Junejo et al. [12] used a tracking algorithm to capture trajectories of people from a surveillance camera, and used a graph-based approach for trajectory clustering. Such clusters should contain “normal” trajectories, and movements with significant discrepancies with respect to “normal” trajectories are classified as suspect motion. Makris and Ellis [14] used the same basic concepts of trajectory clustering, but with a probabilistic approach based on Gaussian Mixture Models.

Fung and Jerrat [8] proposed a system where neural networks are used in order to determine the order in which spatial and pre-defined zones are populated. Then unusual trajectories can be detected if one tracked trajectory does follow one of these orderings. The model described in [15] does not detect directly unusual or usual behaviors. Instead, it determines the locations where people typically go, and it uses a Markov model to detect if patterns different than observed ones happened.

It can be observed that the essence of most references is similar: to detect unusual behavior if the new trajectory differs from a set of trajectories obtained in a training period, according to a similarity measure. Such measure typically takes into account the spatial distance between trajectories and coherence of velocity vectors. In this work, we decouple the spatial information from the “flow coherence” information, and classify trajectories as normal or unusual with respect to these two criteria. The proposed model is described next.

3 The proposed model

The scope of this paper is to detect unusual trajectories automatically, using surveillance cameras. Let us consider that a given environment is monitored by a static camera. Also, let us assume that people move following main direction flows, and occupying roughly the same portion of space within a certain time period. For instance, if we consider a region around a restaurant close to lunch time, people would mostly walk toward the restaurant.

3.1 Automatic people tracking in video sequences

Several vision-based techniques for people tracking in video sequences have been proposed in the past years [3, 4, 6, 16, 18], most of them with surveillance purposes. In these applications, an oblique (or almost lateral) view of the scene is required, so that faces of the individuals can be recognized. In our case, we rely basically on surveillance cameras installed at the top of buildings, generating (approximately) top-view images. It is important to notice that the expected longitudinal projection of a person in oblique-lateral views is explored by several tracking algorithms, such as [3, 6, 9]. However, such hypothesis clearly does not apply in this work, requiring a different strategy for people tracking. In fact, a person's head is a relatively invariant feature in top-view (or almost top-view) camera setups, indicating that tracking can be performed through template matching.

In the proposed approach for people tracking, we adopted a background model that incorporates shadow removal [11] to detect foreground objects (blobs) in grayscale video sequences. Assuming that the head of the person is at the center of the blob, we apply the Distance Transform (DT) to the negative of each new foreground blob (i.e. background pixels are used as the binary image required to compute the DT). The global maximum within each blob returns the approximate location of the head center, and square template T centered at this position is created.

For blob tracking, we compute the Sum of Squared Differences (SSD) between the template T and a small neigh-

neighborhood of T (since there is a maximum displacement for each person in consecutive frames), and retrieve the position that minimizes the SSD as the new template center. Such correlation procedure is repeated for all subsequent frames, until the person disappears from the camera view.

Although the head is a good choice for the correlation template, head tilts and illumination changes may vary the graylevels within the template. Also, the procedure for selecting the initial template may not detect exactly the center of the head. To cope with such situations, T is updated every M_f frames (we used $M_f = 5$ for sequences acquired at 15 FPS). As a result of the tracking procedure, we get the equations $(x_i(t), y_i(t))$, for $i = 1, \dots, N$, where N is the number of tracked people. These trajectories are used to compute a Spatial Occupancy Map and groups of coherent trajectories in the observed period, as explained next.

3.2 Spatial Occupancy Maps

A Spatial Occupancy Map (SpOM) is an image with the same size as the *Region of Interest* captured by the camera (or ROI, which is assumed to be rectangular), that provides the average occupancy of each spatial position in the training region at a given scale. The underlying idea behind the SpOM is simple: initialize the SpOM with zeros, and increment each position of the SpOM (at each frame) every time that a person is detected at this position during the training period. At the end of this procedure, the SpOM would contain large values at highly occupied positions. However, this approach would also lead to a highly local analysis, since one could be able to see individual trajectories. Also, tracked trajectories are one pixel wide (they relate to the “center” of the tracked person), while the space occupied by the person is larger. Furthermore, tracking algorithms typically are not pixelwise accurate, so that two persons walking exactly the same path could lead to two different trajectories.

To cope with these issues, we include a scale parameter σ when building the SpOM. Let us consider a certain person located at the position (x, y) at a given frame. Instead of incrementing only the position (x, y) of the SpOM, we spread the influence of this point to a neighborhood according to a Gaussian weight function with standard deviation σ . Larger values of σ lead to a wider influence of each trajectory, corresponding to a coarser scale of analysis; on the other hand, smaller values of σ lead to better discrimination of individual trajectories, resulting in a finer (more local) scale of analysis.

Mathematically, the SpOM S_σ at the scale σ is given by:

$$S_\sigma(x, y) = \sum_{i=1}^N \sum_{t=1}^{N_f(i)} g_\sigma(x - x_i(t), y - y_i(t)), \quad (1)$$

where N is the number of tracked people, $N_f(i)$ is the duration (in frames) of the i^{th} trajectory, and $g_\sigma(x, y)$ is a truncated discrete bidimensional gaussian kernel, given by:

$$g_\sigma(x, y) = \begin{cases} \frac{1}{c} e^{-\frac{x^2+y^2}{2\sigma^2}}, & \text{if } -2\sigma \leq x, y \leq 2\sigma \\ 0 & \text{otherwise} \end{cases}, \quad (2)$$

and c is a normalization constant so that $\sum_{(x,y)} g_\sigma(x, y) = 1$.

A simple and fast way to implement Equation (1) is through the following convolution:

$$S_\sigma(x, y) = S_\delta(x, y) * g_\sigma(x, y), \quad (3)$$

where S_δ is the SpOM obtained by incrementing the value $S_\delta(x, y)$ every time that a person is detected at position (x, y) , at each frame. This is exactly the same as computing Equation (1) using $g_\sigma(x, y)$ as a discrete Dirac delta function, instead of a Gaussian kernel.

Figure 1 illustrates the computation of SpOMs using different values for σ . Figure 1(a) shows the filmed environment, and Figure 1(b) shows individual tracked trajectories. The computed SpOMs using $\sigma = 5$ and $\sigma = 15$ are illustrated, respectively, in Figures 1(c) and 1(d). In general, σ should be chosen based on the average size of a person in a given camera setup. Considering that in top-view images each person is roughly an ellipse, an appropriate choice for σ is the largest semi-axis of the approximating ellipse. For the camera setup in Figure 1(a), $\sigma = 15$ is an appropriate choice.

3.2.1 Trajectory Occupancy (SpOM Test)

Let us consider the SpOM $S_\sigma(x, y)$, for a given value σ . Let $(x_i(t), y_i(t))$ denote the trajectory of person i to be analyzed, where t denotes time. For usual trajectories (with respect to the spatial occupancy criterion), the SpOM $S_\sigma(x, y)$ should be sufficiently large along the tracked curve $(x_i(t), y_i(t))$. In fact, the function $S_{\sigma,i}(t) = S_\sigma(x_i(t), y_i(t))$, for $0 \leq t \leq N_f(i)$, represents the spatial occupancy of person i across time. Such function can be used to detect portions of the trajectory that present low spatial occupancy, by comparing $S_{\sigma,i}(t)$ to a threshold T_{som} : frames t such that $S_{\sigma,i}(t) \leq T_{\text{som}}$ are considered “unusual” with respect to the spatial occupancy.

A very simple choice would be to set $T_{\text{som}} = 0$, so that every non-zero pixel of $S_\sigma(x, y)$ belongs to a valid occupied region. However, the convolution with the Gaussian kernel used to compute $S_\sigma(x, y)$ tends to increase the region that was actually occupied by tracked people in the training period. Our suggestion is to obtain the threshold T_{som} adaptively from the SpOM by removing a portion r of the smallest values of $S_\sigma(x, y)$ (such discarded values are

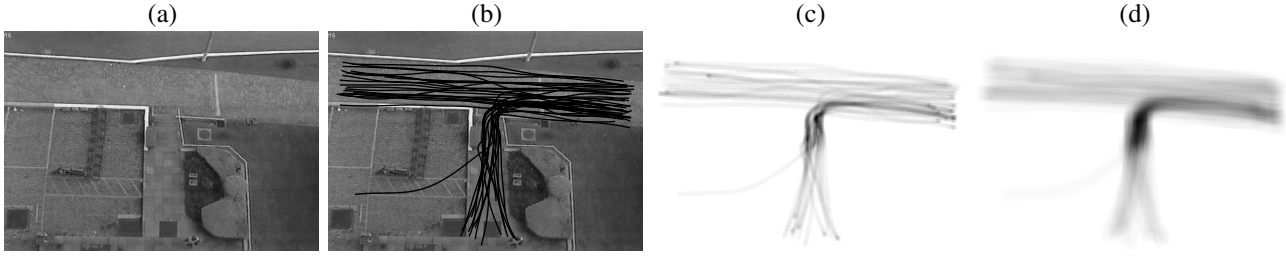


Figure 1. (a) Filmed environment. (b) Individual trajectories. (c) SpOM computed with $\sigma = 5$. (d) SpOM computed with $\sigma = 15$

associated with the spread produced by the tail of the Gaussian), as described next.

Let Q be a vector containing all the non-zero values of the SpOM S_σ , and let Q_s represent the sorting (in ascending order) of Q . If n is the length of Q_s , then:

$$T_{\text{som}} = Q_s(\lfloor rn \rfloor), \quad (4)$$

where $\lfloor \cdot \rfloor$ represents the integer part of a number. In other words, Equation (4) retrieves the value of Q that lies in the r -percentile of its distribution. In all experiments, we used $r = 0.4$.

Figure 2(a) illustrates a tracked trajectory in the test period superimposed to the SpOM computed in the training period. The plot of the corresponding function $S_{\sigma,i}(t)$ is shown in Figure 2(b)¹. It can be observed that the tracked person presents two portions in which the spatial occupancy is very small (in fact, it is zero in a considerable fraction of these portions), meaning that he/she walked on a region that was not occupied in the training period.

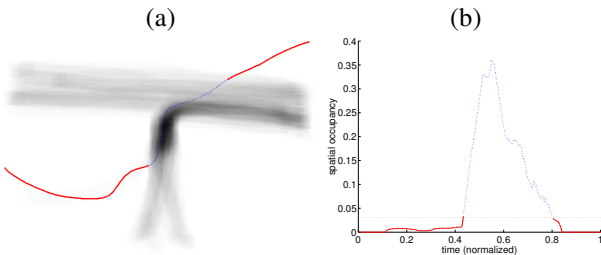


Figure 2. (a) Trajectory overlaid to the SpOM. (b) Spatial occupancy along the trajectory shown in (a).

3.2.2 Trajectory Distance (DT Test)

¹For visualization purposes, we normalized t such that $0 \leq t \leq 1$ in all trajectories.

A relatively common situation occurs when a person walks very close (or a little outside) the occupied region in the training period. In such case, the associated function $S_{\sigma,i}(t)$ could be very small (or even null) during a large portion of the trajectory, which would lead to its classification as unusual. An example of such trajectory is illustrated in Figure 3.

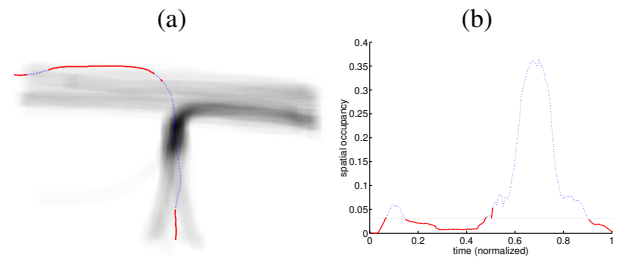


Figure 3. (a) Trajectory overlaid to the SpOM. (b) Spatial occupancy along the trajectory shown in (a).

Also, if a person is moving along a path that was not occupied in the training period, the corresponding function $S_{\sigma,i}(t)$ would be identically zero, regardless of the distance from the person to the occupied region. One way to evaluate the distance of a given path to the “occupied” region is use the Distance Transform.

The Distance Transform (DT) is a well known tool in computational geometry and image processing. Given a binary image I , the distance transform is an image D with the same size as I , such that its value $D(x, y)$ at a certain pixel (x, y) corresponds to the shortest distance between (x, y) and any non-zero pixel of I .

To use the DT, we must initially build a binary image where pixels marked as one represent valid occupied regions. For that purpose, we apply a threshold to the pre-

viously computed SpOM image $S_\sigma(x, y)$:

$$I(x, y) = \begin{cases} 1 & \text{if } S_\sigma(x, y) \geq T_{\text{som}} \\ 0 & \text{otherwise} \end{cases}, \quad (5)$$

where T_{som} is exactly the same threshold used to detect portions of trajectories with low spatial occupancy, as described in Section 3.2.1. The binarized version of the SpOM shown in Figure 1(d) is illustrated in Figure 4(a), and the corresponding Distance Transform is illustrated in Figure 4(b).



Figure 4. (a) Binarized SpOM. (b) Distance Transform of (a).

Let $D(x, y)$ denote the Distance Transform of the binary image $I(x, y)$. The evolution of the minimum distance from a trajectory $(x_i(t), y_i(t))$ and the occupied region (represented by the binary image I) is given by:

$$d_i(t) = D(x_i(t), y_i(t)), \quad (6)$$

and unusual portions of the trajectory are detected when $d_i(t) > T_{\text{dist}}$, where T_{dist} is the maximum allowed distance from the trajectory to the valid occupied region. Although we believe that T_{dist} is context-dependent, a default value could be automatically set based on σ (which, in turn, was chosen based on the average size of a person in image coordinates, as described in Section 3.2). In all examples shown in this work, we used $T_{\text{dist}} = 2\sigma$.

The “distance” test (or simply DT test) applied for the trajectories illustrated in Figures 2 and 3 are shown, respectively, in Figures 5 and 6. As expected, the trajectory in Figure 5 presents portions with large deviations from the valid occupied region (hence, marked as unusual). On the other hand, the trajectory in Figure 6 is always close to the valid occupied region, and was marked as usual at all positions².

It is interesting to notice that the plots $S_{\sigma,i}(t)$ and $d_i(t)$, used in the SpOM test and DT test, respectively, provide complementary information. The function $S_{\sigma,i}(t)$ only carries information in valid occupied regions, and zero otherwise. On the other hand, the function $d_i(t)$ carries information only outside valid occupied regions, and zero otherwise. We believe that the DT test is a better choice to detect

²Please, notice that the plots in Figures 5(b) and 6(b) are shown with different scales, but using the same threshold $T_{\text{dist}} = 30$.

unusual portions of the trajectory, and $S_{\sigma,i}(t)$ can be used as an auxiliary tool to evaluate the mean occupation of the trajectory.

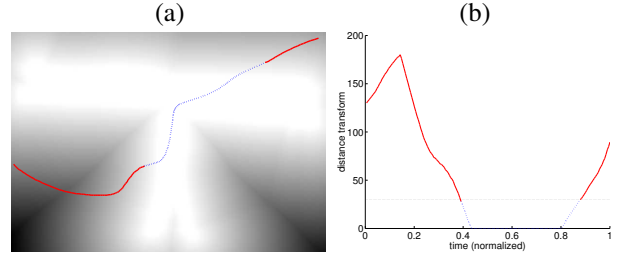


Figure 5. (a) Trajectory overlaid to the Distance Transform. (b) Distance from a valid occupied region along the trajectory shown in (a).

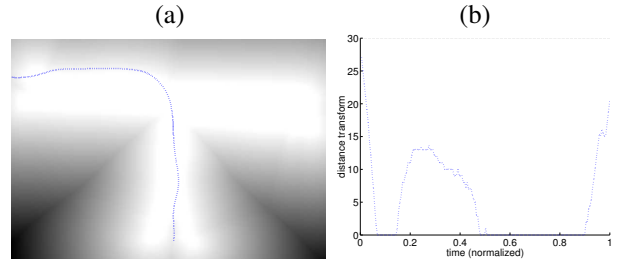


Figure 6. (a) Trajectory overlaid to the Distance Transform. (b) Distance from a valid occupied region along the trajectory shown in (a).

So far, we have classified individual portions of a trajectory into usual or unusual, using either the SpOM test or the DT test. We can also provide a classification of the whole trajectory, by comparing the duration of unusual portions with the total trajectory duration. If D_{unusual}^i denotes the duration of the unusual portions of the i^{th} trajectory (in frames), the whole trajectory is classified as unusual if:

$$D_{\text{unusual}}^i < T_{\text{unusual}} N_f(i), \quad (7)$$

where $0 \leq T_{\text{unusual}} \leq 1$ is a threshold, and again $N_f(i)$ represents the duration of the trajectory (in frames). Although T_{unusual} is application and context dependent, we used $T_{\text{unusual}} = 0.3$ in all examples. As an illustration, D_{unusual}^i values for the trajectories shown in Figures 5 and 6 are, respectively, 0.62 and 0. The first trajectory is then classified as unusual, while the second one is usual, using the DT criterion.

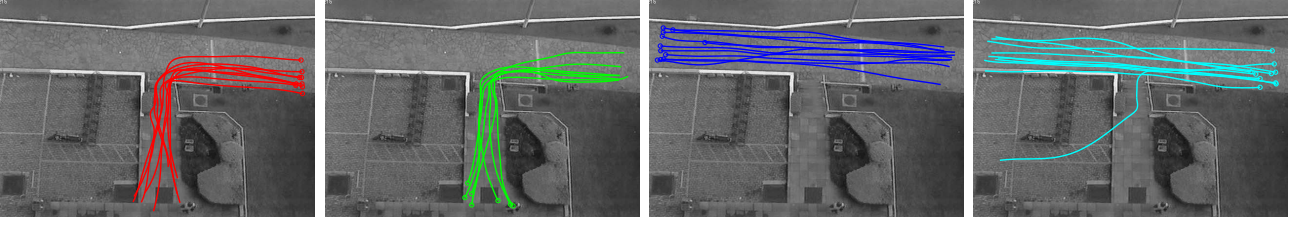


Figure 7. Different “flow clusters” detected for the environment shown in Figure 1(a).

3.3 Trajectory Consistency

The method described above can be used to detect unusual movements with respect to the spatial occupancy of an environment. However, one trajectory may present a large SpOM at all points, but its dynamics may be unusual when compared to other trajectories of the training period. For example, all persons in the observed period may walk in the east-west direction, having a roughly uniform spatial occupation. If a person moves in the same environment in the west-east direction, he/she could present a normal spatial occupancy, but the trajectory would not be coherent with the training set.

Our second criterion for detecting unusual movements is based on the coherence of trajectories, in the sense that two similar trajectories should present roughly the same flow directions. It should be observed that grouping trajectories into similar classes was used in [12–14]. However, all these methods employ some kind of distance-based metric to define similar trajectories, while the present approach intend to cluster trajectories according to main direction flows. For example, two persons moving at coherent speeds, from east to west, at a sufficiently large lateral distance apart, would probably generate different clusters according to the approaches [12–14]. However, these persons present similar flow directions, and would be considered similar in our approach. Our trajectory clustering algorithm is described next.

Let $(x^n(s), y^n(s))$, for $0 \leq s \leq 1$, be a reparametriza-

tion of the trajectory $(x(t), y(t))$ with respect to the normalized arclength s , so that $(x^n(0), y^n(0))$ represents the initial point of the trajectory and $(x^n(1), y^n(1))$ represents the end point. Such trajectory is characterized by a set of N displacement vectors $(\Delta x_i, \Delta y_i)$ computed at equidistant arclengths:

$$(\Delta x_i, \Delta y_i) = (x(s_{i+1}) - x(s_i), y(s_{i+1}) - y(s_i)), \quad (8)$$

where $s_i = \frac{i}{N}$, $i = 0, \dots, N - 1$. For each trajectory j , a $2N$ -dimensional feature vector \mathbf{f}_j is obtained by combining the N displacement vectors associated with the trajectory:

$$\mathbf{f}_j = (\Delta x_0, \Delta y_0, \Delta x_1, \Delta y_1, \dots, \Delta x_{N-1}, \Delta y_{N-1}). \quad (9)$$

Coherent trajectories are expected to produce similar feature vectors \mathbf{f} , generating a cluster in the $2N$ -dimensional space, which is modeled as a Gaussian probability distribution. Each class k is characterized by its mean vector $\boldsymbol{\mu}_k$ and covariance matrix \mathbf{C}_k , as well as the prior probability w_k . Since each cluster relates to a different Gaussian function, the overall distribution $p(\mathbf{x})$ considering all feature vectors \mathbf{f} can be modeled as a mixture of Gaussians:

$$p(\mathbf{x}) = \sum_{k=1}^{k_{\max}} w_k p_k(\mathbf{x}), \quad (10)$$

where k_{\max} is the number of classes and

$$p_k(\mathbf{x}) = \frac{(2\pi)^{-N}}{\sqrt{|\mathbf{C}_k|}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_k)^T \mathbf{C}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \right\} \quad (11)$$

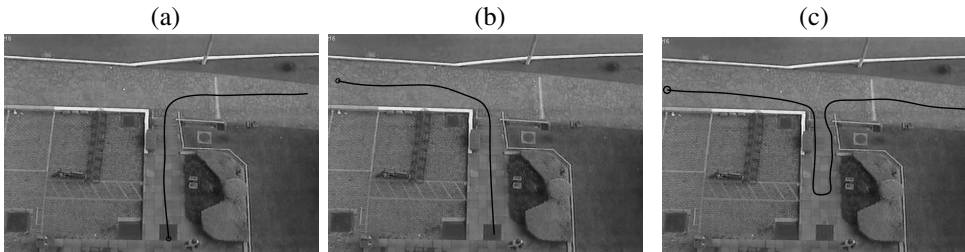


Figure 8. Examples of usual (a,c) and unusual (b) trajectories with respect to trajectory consistency, using the clusters displayed in Figure 7.

is the normal distribution associated with the k^{th} class. The number of Gaussians in the mixture (which corresponds to the number of clusters), as well as the distribution parameters of each individual distribution can be obtained automatically using the unsupervised clustering algorithm described in [7]. Then, each feature vector \mathbf{x} in the training period is assigned to a class k according to:

$$k = \underset{j \in \{1, \dots, k_{\max}\}}{\operatorname{argmin}} w_j p_j(\mathbf{x}), \quad (12)$$

which is equivalent to selecting the class with the highest *a posteriori* probability, according to Bayes rule.

The number N of displacement vectors used to assemble \mathbf{f}_j is chosen based on how structured the flow of people is. For relatively simple trajectories, small values of N can capture the essence of the trajectories. On the other hand, more complicated trajectories (with many turns) are better characterized using larger values of N . In general, public spaces tend to present main flow directions, and $N = 1$ or $N = 2$ are usually good choices. It should also be noticed that larger values for N result in higher dimensional feature vectors, requiring a larger number of sample for the clustering algorithm. For example, Figure 7 illustrates the clustering procedure for the environment shown in Figure 1. As it can be observed, four main “direction flows” were detected by the clustering algorithm, using $N = 1$.

During the clustering process (in the training period), we can also determine a minimum acceptable probability threshold P_k for each class:

$$P_k = \min_{\mathbf{x} \in \text{class } k} \{p_k(\mathbf{x})\}. \quad (13)$$

In the test period, the feature vector \mathbf{y} related to a new trajectory is initially assigned to an existing cluster k according to Equation (12), but it is rejected (hence, considered unusual) if its probability is lower than the minimum acceptable for the class, i.e., if:

$$p_k(\mathbf{y}) < P_k. \quad (14)$$

Figure 8 illustrates usual and unusual trajectories with respect to trajectory consistency. The trajectory in Figure 8(a) was associated with the second cluster (second image in Figure 7, from left to right), and was not rejected by condition (14). Consequently, it was considered an usual trajectory. On the other hand, the trajectory in Figure 8(b) was initially assigned to the third cluster, but it satisfied condition (14), and was rejected (hence, considered unusual). Figure 8(c) shows an example of possible misclassification with respect to the trajectory coherence criterion (using $N = 1$ for clustering). In this example, a person climbed up the stairs and right after climbed down, and this movement may be considered unusual. However, the trajectory was considered usual (it was associated with the third cluster in

Figure 7 and was not rejected by condition (14)). Such misclassification happened because the value $N = 1$ captures only the global displacement of the trajectory, and positions in the middle of the trajectory (where the possible suspect motion occurred) are disregarded. However, this unusual behavior in the middle of the trajectory can be detected using $N = 2$ in the clustering stage. It is interesting to notice that all three trajectories in Figure 8 were considered usual with respect to spatial occupancy (in the DT test).

4 Experimental Results

In this Section we present some results of usual/unusual trajectory detection based on spatial occupancy and trajectory coherence. Figure 9(a) shows the trajectory of a tracked person. Usual portions of the trajectory (according to the DT criterion) are shown in blue, and unusual portions in red. It can be observed that the transition from usual to unusual was detected on-the-fly, so that security staff could be alerted. The whole trajectory was classified as unusual in the DT test, according to Equation (7). This trajectory is also unusual using the consistency criterion, since it is assigned to class 3 and satisfies condition (14). Indeed, this trajectory may be considered suspect, since the person left the pathway and walked on the grass.

Figure 9(b) illustrates the trajectory of another tracked person, along with its usual/unusual portions (in the DT test). It can be observed that this person started on the pathway and deviated to the parking lot (non-occupied region in the training period), approaching again to the pathway, and finally leaving the scene. This trajectory was also considered unusual with respect to the cluster coherence criterion, since it was assigned to class 4 and exceeds the threshold as in condition (14). The attached short movie shows the actual footages used in Figures 9(a) and 9(b), along with tracked trajectories and on-the-fly detection of unusual trajectory portions using the DT criterion.

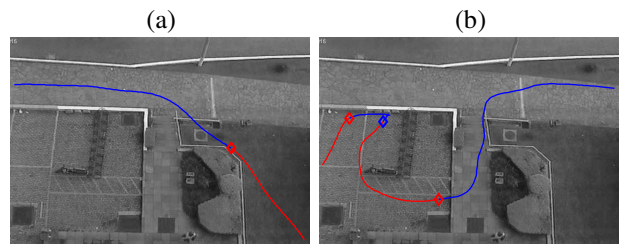


Figure 9. Tracked trajectories with usual (blue) and unusual (red) portions, using the DT criterion.

5 Conclusions

This paper presented a new approach for classifying trajectories as usual or unusual using computer vision, based on two criteria: spatial occupancy and trajectory coherence. In a training period, tracking algorithms are applied to obtain individual trajectories. Such trajectories are used to build a Spatial Occupancy Maps (SpOM), that provides a representation of regions covered by tracked trajectories. Also, trajectories are clustered into coherent classes, representing main “direction flows”. Then, a new trajectory is compared both with the SpOM and the clusters. Comparisons with the SpOM provide portions of the trajectory that lie in low-occupancy regions, and also the deviation from the person to a valid region. Comparisons with the clusters provide the coherence of the new trajectory with respect to the training period.

The proposed method can be used as an auxiliary tool in surveillance systems. For instance, the system could trigger an alarm when a tracked person exceeds the threshold T_{dist} , meaning that he/she is far enough from the valid occupied region to be considered in an unusual position. The human operator could then take a better look at that person.

It is important to emphasize that our method relies strongly on the trajectories used in the training period, which are examples of usual motion. Such training period may vary depending on the hour of the day, and could lead to different time-dependent training periods. For example, it is common to see people moving to the parking lot to get their cars right after classes, but it is not an usual behavior in the middle of classes.

For future work, we intend to explore the issue of time-dependent training periods, and include the relationships among people in our analysis.

Acknowledgements

This work was developed in collaboration with HP Brazil R&D.

References

- [1] B. A. Boghossian and S. A. Velastin. Motion-based machine vision techniques for the management of large crowds. *Electronics, Circuits and Systems, 1999. Proceedings of ICECS '99. The 6th IEEE International Conference*, 2:961 – 964, September 1999.
- [2] H. Buxton and S. Gong. Advanced visual surveillance using bayesian networks. In *International Conference on Computer Vision*, Cambridge, Massachusetts, June 1995.
- [3] F. Cheng and Y. Chen. Real time multiple objects tracking and identification based on discrete wavelet transform. *Pattern Recognition*, 39(6):1126–1139, June 2006.
- [4] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati. Detecting moving objects, ghosts, and shadows in video streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1337–1342, October 2003.
- [5] A. Davies, J. H. Yin, and S. Velastin. Crowd monitoring using image processing. *IEE Electronic and Communications Engineering Journal*, 7(1):37–47, February 1995.
- [6] A. Elgammal, R. Duraiswami, D. Harwood, and L. Davis. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. *Proceedings of the IEEE*, 90(7):1151–1163, 2002.
- [7] M. A. T. Figueiredo and A. K. Jain. Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):381–396, March 2002.
- [8] C. C. Fung and N. Jerrat. A neural network based intelligent intruders detection and tracking system using cctv images. In *Region 10 Conference on Intelligent Systems and Technologies for the Next Millennium (Tencon' 2000)*, 2000.
- [9] I. Haritaoglu, D. Harwood, and L. Davis. W4: Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):809–830, August 2000.
- [10] W. Hu, T. Tan, L. Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on Systems, Man, and Cybernetics - Part C*, 34(3):334–352, August 2004.
- [11] J. C. S. Jacques Jr., C. R. Jung, and S. R. Musse. A background subtraction model adapted to illumination changes. In *IEEE International Conference on Image Processing*, 2006.
- [12] I. Junejo, O. Javed, and M. Shah. Multi feature path modeling for video surveillance. In *Proceedings of International Conference on Pattern Recognition*, pages II: 716–719, 2004.
- [13] J. Lou, Q. Liu, T. Tan, and W. Hu. Semantic interpretation of object activities in a surveillance system. In *Proceedings of the IEEE International Conference on Pattern Recognition*, volume 3, pages 777–780, August 2002.
- [14] D. Makris and T. Ellis. Learning semantic scene models from observing activity in visual surveillance. *IEEE Transactions on Systems, Machines and Cybernetics - Part B*, 35(3):397–408, June 2005.
- [15] N. T. Nguyen, H. H. Bui, S. Venkatesh, and G. West. Recognising and monitoring highlevel behaviours in complex spatial environments. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 620–625, June 2003.
- [16] H. Ning, T. Tan, L. Wang, and W. Hu. People tracking based on motion model and motion constraints with automatic initialization. *Pattern Recognition*, 37(7):1423–1440, July 2004.
- [17] C. Stauffer, W. Eric, and L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):747 – 757, August 2000.
- [18] Y. Tian, M. Lu, and A. Hampapur. Robust and efficient foreground analysis for real-time video surveillance. In *IEEE Computer Vision and Pattern Recognition*, pages I: 1182–1187, 2005.