Advanced Video Surveillance with Pan Tilt Zoom Cameras

Rita Cucchiara, Andrea Prati, Roberto Vezzani
University of Modena and Reggio Emilia - Italy

Abstract

In this paper an advanced video surveillance system is proposed. Our goal is the detection of the people's heads to allow their obscuration for privacy issues or to perform recognition tasks. We propose a system based on active PTZ (Pan-Tilt-Zoom) cameras that produce head images having a large enough size, and can cover an area larger than still cameras. Since conventional approaches are not suitable to PTZ cameras, the proposed approach is based on the so-called direction histograms to compute the ego-motion and on frame differencing for detecting moving objects. It exploits post-processing and active contours to extract precise shape of moving objects to be fed to a probabilistic algorithm to track moving people in the scene. Person following, instead, is based on simple heuristic rules that move the camera as soon as the selected person is close to the border of the field of view. Finally, a color and shape based head detection that takes advantage of the people tracking is presented. Experimental results on a live active camera demonstrate the feasibility of real-time person following and of the consecutive head detection phase.

1. Introduction

This paper proposes a system of advanced video surveillance for moving people segmentation, tracking and face detection from a moving camera. The method is designed to work in real time for creating a mosaic image of the whole scene (by registering overlapped images provided by successive frames of the active camera), detect and track moving people very quickly, and follow a selected person. *Person following* is intended as the task with which the system keeps the person framed by the current view by automatically moving the active camera. Finally, the heads of the followed people are extracted to allow different kind of surveillance tasks, such as face obscuration (for applications subjected to privacy issues), head following, face recognition, snapshot logging for post-analysis and information retrieval, and so on.

We propose a new method for fast ego-motion computation based on the so-called *direction histograms*. The method works with an uncalibrated camera that moves with an unknown path and it is based on the compensation of the camera motion (i.e., the *ego-motion*) to create the mosaic image and on the frame differencing to extract moving objects. Successive steps eliminate the noise and extract the complete shape of the moving objects in order to exploit an appearance-based probabilistic tracking algorithm. Person following, instead, is based on a quite intuitive method that moves the camera when the person is near to exit from the field of view of the camera.

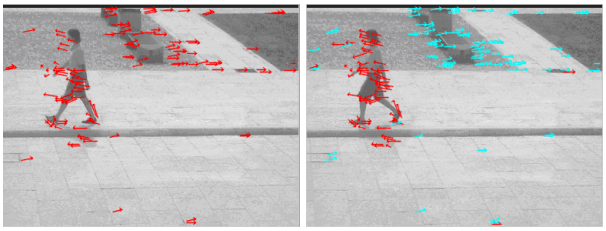
The segmentation of moving objects becomes more critical when the video is acquired by a moving camera with an unconstrained and a priori unknown motion. Proposals from single camera can be grouped into three classes: based on ego-motion computation, based on motion segmentation, and based on region merging with motion. The approaches in the first class aim at estimating the camera motion (or ego-motion) through the evaluation of the dominant motion with different techniques and models in order to obtain compensated videos and to apply algorithms developed for fixed camera (frame differencing, as in [3], or background suppression, as in [11]). In [8] Kang *et al.* define an adaptive background model that takes into account the camera motion approximated with affine transformation. Tracking of moving object is achieved by means of a joint probability data association filter (JPDAF). In methods based on motion segmentation the objects are mainly segmented by using the motion vectors computed at pixel level ([9]). The vectors are then clustered to segment objects with homogeneous motion. Finally, the approaches based on region merging with motion are hybrid approaches in which the objects are obtained with a segmentation based on visual features, and next merged on motion parameters computed on a region-level [4]. It is worth noting that most of the reported approaches are computationally very expensive and cannot meet real-time constraints (and those that meet them use either special-purpose devices or a set of limiting assumptions).

We propose a real-time approach for advanced video surveillance capable to exploit an active camera to follow a person and, simultaneously, extract his head for obscuring it or for zooming on it for recognition purposes. The next sections will describe the algorithms used for people tracking from an active camera and for head detection.

2. People Tracking from Active Camera

Our approach for moving object segmentation from moving camera consists in two basic steps: first, the ego-motion is estimated and compensated to build a mosaic image and, second, frame differencing and post-processing are applied to extract the single moving objects.

The motion vectors of the current frame are extracted using a pyramidal implementation of the Lucas-Kanade algorithm (Fig. 1(a)). Then, they are clustered to find the dominant motion, that corresponds to the ego-motion assuming that the background is dominant over the moving objects.



(a) Motion vectors with the Lucas-Kanade algorithm (b) Clustered motion vectors

Figure 1: Example of extraction and clustering of the motion vectors

The clustering is performed with an innovative and fast process. It can be demonstrated that, for small pan and tilt angles the camera motion model can be approximated with a pure translational model. With this hypothesis, a *direction histogram* containing all the directions of the extracted motion vectors is built (see Fig. 2(a)). Let $\overrightarrow{\mathbf{mv}}(x,y) = (\rho(x,y), \alpha(x,y))$ be the motion vector computed at the coordinate (x,y), where ρ and α are the magnitude and the angle of the vectors expressed using the polar coordinates. We define the direction histogram $DH(\beta)$ as in Equation 1.

$$DH(\beta) = \#\left\{\overrightarrow{\mathbf{mv}}(x,y) | \alpha(x,y) = \beta\right\} \tag{1}$$

with β ranging from 0 to 2π. A 1-D Gaussian filter $G(\mu,\sigma)$ centered on the histogram peak μ is applied on the histogram to eliminate motions different from the dominant one as in Equation 2.

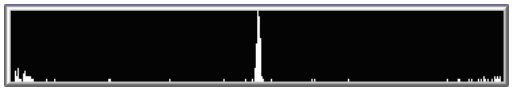
$$\widetilde{DH}(\beta) = DH(\beta) \cdot G(\mu,\sigma) \tag{2}$$

where $\mu = \arg\max_{\beta} DH(\beta)$ and σ is a parameter set to 1 for most of the experiments.

The resulting histogram $\widetilde{DH}(\beta)$ of Equation 2 (shown, for instance, in Fig. 2(b)) allows to divide the motion vectors into two groups, one due to the camera motion (in cyan in Fig. 1(b)) and one due to moving objects (in red in Fig. 1(b)), and to compute the direction $\overline{\alpha}$ and amplitude $\overline{\rho}$ of the ego-motion by averaging the vectors retained by the Guassian filter as in Equation 3.

$$\overline{\alpha} = \arg\max_{\beta} \widetilde{DH}(\beta) \quad ; \quad \overline{\rho} = \frac{\sum_{\rho \in R} \rho}{|R|} \tag{3}$$

where $R = \left\{\rho(x,y) | \widetilde{DH}\left(\alpha(x,y)\right) \neq 0\right\}$.



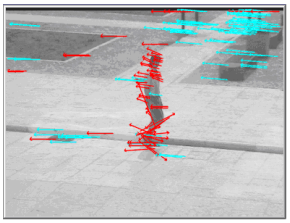
(a) Direction histogram

(b) Filtered direction histogram

Figure 2: Direction histograms before and after the application of gaussian filter

This approach, though intuitive and simple, has proven to act very well, given that the above-mentioned hypothesis holds. For example, Fig. 3 reports the result in the case of a person moving with motion concordant with the camera. It is worth noting that the direction histogram (Fig. 3(a)) contains two peaks corresponding to the ego-motion and the person, respectively. However, Fig. 3(b) shows that, though some errors are present, accuracy is still acceptable.

(a) Direction histogram

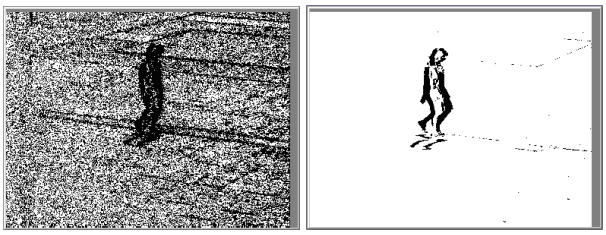


(b) Motion vectors with LK

Figure 3: Result of the motion vector clustering in the case of a person moving in the same direction of the camera.

Once the ego-motion is estimated, the current frame is registered (assuming a translational motion model) by compensating for the camera motion given by the vector $(\overline{\rho}, \overline{\alpha})$.

The difference in the results performing frame differencing before and after the compensation is shown in Fig. 4. Moving pixels are indicated in black.



(a) Without compensation (b) With compensation

Figure 4: Frame differencing (a) without and (b) with egomotion compensation

As evident in Fig. 4(b), the result provided by frame differencing are still far from being optimal, for both the noise due to imprecise image registration and the ghost of the moving objects. For this reason, post-processing steps must be used. Noise and small areas are removed by morphological operations, whereas ghosts are eliminated by merging information provided by a connected-components analysis and by a Canny edge detector: only edges with at least one point (in the 3x3 neighborhood) detected as moving are retained. Fig. 5(a) shows an example of retained edges.

Based on these information, the single moving objects are located. Their shape, however, is imprecisely extracted. Since the performance of the tracking algorithm heavily depends on the precision of the object's shape, a successive step is required. Since standard background suppression techniques are not suitable with our requirements (unknown path, uncalibrated camera, and real-time constraints), we employ a variant of the classical *active contours*, in which the energy is obtained with the following equation:

$$E_i = E_{cont,i} + \frac{E_{curv,i}}{2} + E_{dist,i} \tag{4}$$

Given $p_1, ..., p_n$ a discrete representation of the contour/shape to be modelled, $E_{cont,i}$ represents the contour continuity energy and is set to:

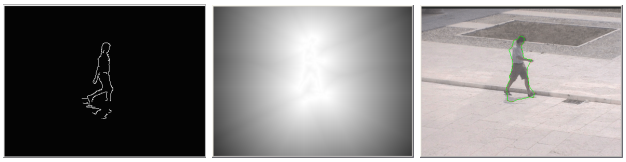
$$E_{cont,i} = \left|\overline{d} - |p_i - p_{i-1}|\right| \tag{5}$$

where $\overline{d}$ is the average distance between each consecutive pair of points. Minimizing this energy means to have the points more equidistant.

$E_{curv,i}$ is the contour curvature energy (the smoother the contour is, the lower the energy) and is defined as:

$$E_{curv,i} = ||p_{i-1} - 2p_i + p_{i+1}||^2 \tag{6}$$

As external energy, we modify the original proposal by considering the image obtained by applying the Distance transform to the image containing the edges retained by the post-processing. Examples of input edge image, external energy image and resulting snake are reported in Fig. 5. Finally, contour filling is employed to obtained a rough segmentation of the person's shape to be provided to the appearance-based probabilistic tracking proposed in [2], that is meant to be robust to occlusions.



(a) Edges (b) Energy (c) Snakes

Figure 5: Active contours: input edge image, external energy image and the resulting snake

Final mosaic image is constructed by superimposing the registered image on the mosaic and applying a simple alpha blending algorithm. Moreover, moving objects are not pasted onto the mosaic.

Once moving people are detected the system allows to select a single person to be followed, by moving the camera to keep him framed. In the current implementation of the system the "youngest" person (in the sense of that tracked by less time) in the scene is followed until he is visible. When he exits from the area the camera either follows the next "youngest" person, if any, or goes to a predefined position. Person following is achieved by moving the camera towards the person as soon as he approaches the limit of the current field of view.

3 Head detection

Face detection is a widely explored research area in computer vision. Two recent surveys ([12] and [5]), collect a large number of proposal about face detection. Most of them are based on a skin color detection [7] followed by a face candidate validation achieved exploiting geometrical and topological constraints. Hsu *et al.* [6], for example, propose a face detection algorithm for color images in the presence of varying light conditions, based on a lighting compensation technique and a non linear color transformation. They detect skin regions over the whole image, and then they generate face candidates imposing spatial constraints. Unfortunately, most of the color-based approaches are very expensive from the computational point of view and it is impossible to perform an accurate face detection at every frame in a real time video surveillance application. To solve this problem, the face detection can be performed only when a new person enters the scene and then adopt a

Figure 6: The elliptical face model adopted for the head detection. The size ratio and the orientation are fixed.

face tracking as the one proposed in Birchfield [1]. A different approach, instead, is the one proposed by Maio and Maltoni in [10] that works on grey scale images. In particular, the face candidates are obtained through an ellipse detection applied over the gradient. The algorithm we designed in this work is based on the elliptical approximation of the head shape and the generalized Hough transform. In particular, the method can be considered as a mixture of the two proposals reported in [1] and [10].

The proposed approach has other important characteristics, that make it particularly suitable for video surveillance applications. As first, it is featured by a low computational cost, which is a mandatory requirement to reach real time performance. Secondly, the images taken from video surveillance systems usually have a lower quality than a picture and the head sizes are smaller since the field of view is kept large to cover a wider area. In such a situation it is impossible to identify face features like eyes, mouth, lips, and so on. Thus, a feature based face detector is not employable, while algorithm based on color and shape are still valid. Finally, the proposed method is able to detect the head of the people turned back. For this reason we prefer to call the implemented algorithm "head detection" instead of "face detection".

3.1 Head model

As above mentioned, in surveillance applications the head size is too small to detect face features like eyes, lips, and so on; thus we adopt a head model based on the color histogram and the border shape only. As first, we exploit an elliptical head model with a fixed size ratio empirically sets to 1.2. Furthermore, we suppose the ellipse to be vertical; in such a manner the ellipse containing the head has three degrees of freedom that could be expressed with the coordinates of the center (X_c, Y_c) and the size W of the horizontal axis (see Figure 6).

In addition to the shape, the color information is used to characterize the head. In particular, we adopt as descriptor the color histogram $\tilde{H}$ computed over the set of points internal to the above defined ellipse. To speedup the detection phase and to reduce the amount of memory required for each model, we employ the compressed space proposed in [1], which is composed by the three components (c_1, c_2, c_3) of Equation 7.

$$\begin{aligned} c_1 &= B - G \\ c_2 &= G - R \\ c_3 &= R + G + B \end{aligned} \tag{7}$$

We use 3 bits for representing c_1 and c_2 (chrominance components) and 2 bits for c_3 (luminance component), so that the color histograms are composed by 256 color bins.

These histograms contain both face and hair colors; thus, a general and unique model can not be obtain, but a custom histogram $\tilde{H}_i$ has to be computed and saved for each person i whenever he enters the scene.

3.2 Color module

Let $HC_i = (X_c, Y_c, W, H)$ be a head candidate for the person i, defined by the coordinates (X_c, Y_c) of the center, the width W and the color histogram H. Given the correspondent head model $\tilde{H}_i$ (i.e., the histogram stored for the same tracked person i), we compute the color-based probability P_C to be a head using the histogram intersection:

$$P_C\left(HC_i \middle| \tilde{H}_i\right) = \frac{\sum_{k=1}^{256} \min\left(H(k), \tilde{H}_i(k)\right)}{\sum_{k=1}^{256} H(k)} \tag{8}$$

From the practical point of view, the histogram intersection gives us a measure of how many colors of the head candidate are present in the reference model. In other words, the probability value is equal to 1 if the candidate is exactly the same of the model; instead it decreases if the candidate contains colors that do not appear in the model.

3.3 Gradient module

Goal of this module is the measure of how much the head candidate has an elliptical shape. To this aim we compute the two normalized gradient maps S_X, S_Y of the image (along the horizontal and the vertical direction respectively) using the Sobel masks. The gradient based probability P_G is obtained as in the following equation.

$$P_G(HC_i) = \frac{\sum_{p \in \mathrm{E}} \sqrt{S_X^2(p) + S_Y^2(p)}}{|E|} \tag{9}$$

where $|E| = \sum_{p \in \mathrm{E}} 1$.

Figure 7: Input frames where the corresponding search areas have been superimposed. The size of the search area is dependent on the motion of the tracked person.

3.4 Head detection

Given a set HC of head candidates HC_i, we select as current detection the one that maximizes a global score $\Phi(HC_i)$ as in Equation 10.

$$\Phi(HC_i) = \alpha \cdot P_C(HC_i) + (1-\alpha) \cdot P_G(HC_i) \quad (10)$$

The parameter α is used to differently weight the color and the gradient module and should be adapted depending on the particular application or video characteristics. In fact, if the head size is too small, the shape term could be less significant and not so distinctive, since other almost circular objects can be present in the scene. Similarly, the video quality could be so much degraded to avoid the efficacy of the color module.

The set $HC(t)$ of head candidates is obtained starting from the set of heads extracted on the last frame $HC(t-1)$. For each head detected at the previous time step $t-1$, a prediction based on constant velocity and constant size is computed for the current frame t. To take into account scale and direction changes, the set $HC(t)$ also contains head candidates of different size and position. In particular, the position is searched in an area whose dimension is function of the person's velocity. The faster the person moves, the larger the search window is. Instead, the size $W(t)$ of the head is searched within a fixed range around the previous size $W(t-1)$. In Fig. 7 are reported some frames in which the head search windows have been superimposed.

3.5 Integration with the background suppression module

To reduce the computational cost and increase the precision of the detection, we use the foreground blobs extracted with a background subtraction module as input of the head detection subsystem. If the camera is still, a common background subtraction technique can be used in order to obtain a valid foreground region as in Fig. 8.

If the adopted camera is moving, instead, we can apply the mosaicing algorithm described in section 2. In this case, even if we cannot extract a reliable foreground region, we can estimate the bounding box of the people and reduce the head search inside these regions. In Fig. 9 the search area (i.e., the bounding box of the person) obtained with the dynamic mosaicing algorithm and the detected head are highlighted.

(a) Current frame (b) Extracted Foreground

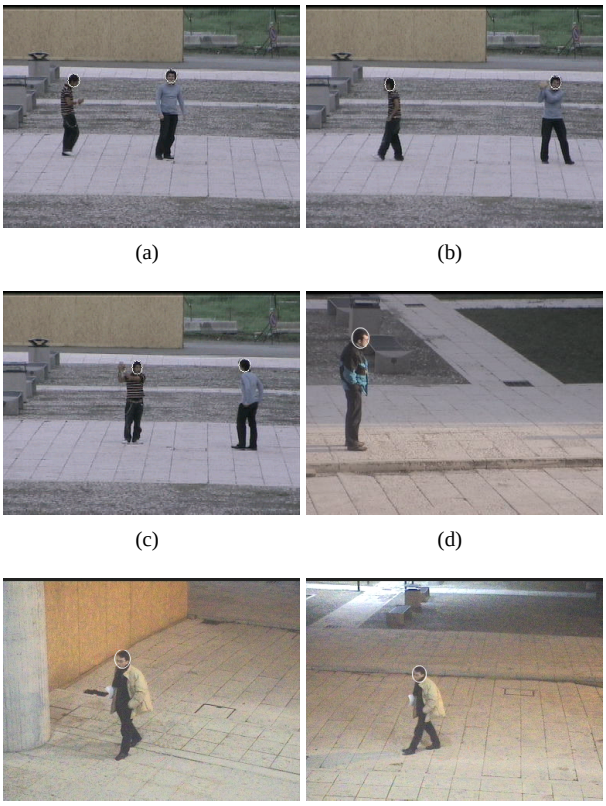
Figure 8: Face detection over the foreground obtained with a background subtraction algorithm

Figure 9: Search area obtained with the dynamic mosaicing algorithm and the detected head

4. Experimental Results

We carried out several tests to check the performances of the described system. We report in Fig. 10 the results of the head detection algorithm obtained with a static camera (Fig. 10(a-c)) and a Pan-Tilt-Zoom one moved with the people following system (Fig. 10(d-f)) described in Section 2. The performances of the system are very encouraging, both in terms of precision and low sensitivity to disturbs (e.g., the hands of the person, that have the same color and similar shape of the head in Fig. 10(b)) and noise (Fig. 10(f)).

The detection of the head enables some interesting applications. For example, the face of the people can be obscured for privacy issues, preventing the recognition of the identity of the monitored people (Fig. 11).



(a) (b) (c) (d) (e) (f)

Figure 10: Example of the face detection output

Conversely, the face could be useful for recognition tasks; in these cases a bigger and more detailed head image can be obtained exploiting the camera zoom. To this aim an auxiliary camera can be adopted in conjunction of the one used for the people following to extract zoomed snapshot of the heads. Otherwise, the same camera can be employed to zoom in on the head region for a while whenever the detected person is still. In Fig. 12 some snapshots automatically taken at different zoom levels are reported.

As described in Section 2, we implemented and tested a dynamic mosaicing algorithm able to extract the foreground region on a moving (Pan-Tilt-Zoom) camera. This evaluation has been carried out by means of both qualitative and quantitative analysis. For qualitative analysis, a large set of videos has been taken with different illumination conditions, different number of people (from none to 5-6 simultaneously moving people), and different movements of the camera (only pan, only tilt, both pan and tilt). This analysis has demonstrated that, if the hypotheses hold, the system produces very good mosaic images, like those shown in Fig. 13. The only distortions appear at the top of the image where they do not affect moving object segmentation. Mosaic images have been also evaluated using a quantitative (i.e., objective) measure such as the PSNR. For example, the PSNR of the mosaic images reported in Fig. 13 with respect to ground truths (generated by exhaustively trying all the possible displacements and choosing that minimizing the error) is 40.82 dB.

Fig. 14 shows a sequence reporting some snapshots of the results for person following. The red bounding box identifies the person followed, while green ones identify other moving objects. The drawings on the bottom right corner of each image show the actual movement of the camera. It is worth noting that these results have been obtained with a completely unsupervised system working on live camera. It is evident that there are some imprecisions: for instance, on row 2, column 3, the second person is not segmented since it is very dark; on row 3, column 2, shadows are connected to the moving person; erroneous moving objects are detected on the column in the last row, columns 3 and 4. In particular, the last snapshot reports a wrong segmentation due to the presence, in the background, of much texture and to the closeup of the scene.

(a) (b)

Figure 11: Results of the “face detection and obscuration” system

(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 12: Results of the “face detection and zoom” system

Figure 13: An example of mosaic image.

From the computational point of view, the system works in real time, with an average frame rate from live camera of about 5 fps, including also the person following task and the following face detection. Considering that the current acquisition device releases 12.5 frames per second, we can properly speak of “real time”.

5. Conclusions

The extraction and tracking of moving objects from a moving camera are difficult tasks, especially under the constraints of unknown camera motion, uncalibrated camera, and fast system’s response. This paper proposes a suitable solution that, given the typical hypotheses of an active surveillance camera, assures a good trade-off between speed and accuracy. Experimental results showed that, if the person does not move too fast with respect to the speed of the camera’s moving head, real-time person following on live camera is feasible.

This advanced surveillance system can be the basis of several applications. In particular, we expand it with a head detection module that has demonstrated good accuracy and that can be used to either simply detect heads (for example, for obscuring them for privacy purposes) or drive the active camera to zoom on the person’s face for further recognition tasks.

6. Acknowledgements

This work was supported by the project L.A.I.C.A. (Laboratorio di Ambient Intelligence per una Città Amica), funded by the Regione Emilia-Romagna, Italy. The authors want also to thank Fabrizio Seghedoni for his help in code fixing and experiments.

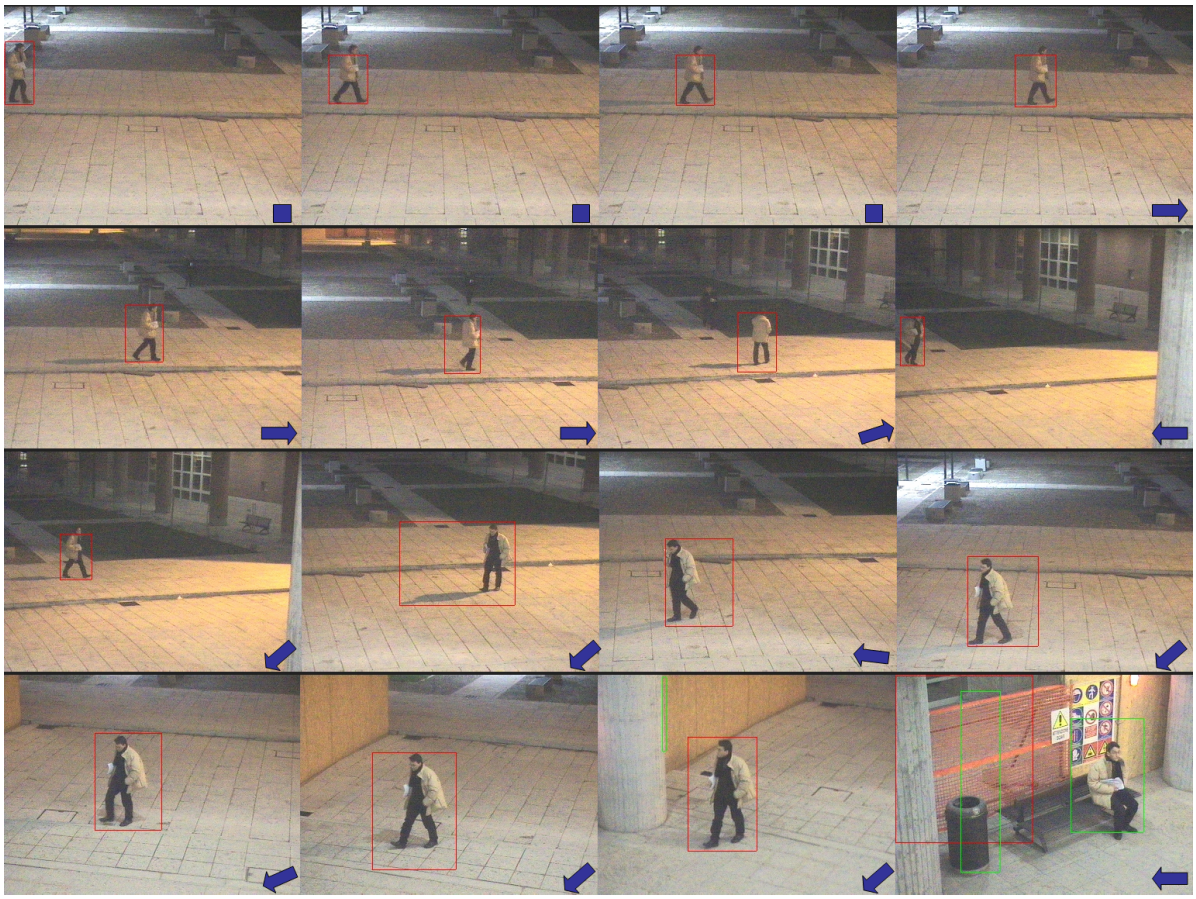


Figure 14: Snapshots from a live sequence with person following.

References

[1] S. Birchfield. Elliptical head tracking using intensity gradients and color histograms. *Proc. of IEEE Computer Vision and Pattern Recognition*, pages 232–237, 1998.

[2] R. Cucchiara, C. Grana, G. Tardini, and R. Vezzani. Probabilistic people tracking for occlusion handling. In *Proceedings of Int'l Conference on Pattern Recognition*, volume 1, pages 132–135, Aug. 2004.

[3] R. Cutler and L. Davis. Robust real-time periodic motion detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):781–796, Aug. 2000.

[4] M. Gelgon and P. Bouthemy. A region-level motion-based graph representation and labeling for tracking a spatial image partition. *Pattern Recognition*, 33:725–740, 2000.

[5] E. Hjelm and B. Low. Face detection: A survey. *Computer Vision and Image Understanding*, 83(3):236–274, 2001.

[6] R. Hsu, M. Abdel-Mottaleb, and A. Jain. Face detection: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):696–706, 2002.

[7] M. Jones and J. Rehg. Statistical color models with application to skin detection. *International Journal of Computer Vision*, 46:81–96, 2002.

[8] J. Kang, I. Cohen, and G. Medioni. Continuous tracking within and across camera streams. In *Proceedings of IEEE Int'l Conference on Computer Vision and Pattern Recognition*, volume 1, pages I–267 – I–272, 2003.

[9] K. Lee, S. Ryu, S. Lee, and K. Park. Motion based object tracking with mobile camera. *Electronics Letters*, 34(3):256–258, Mar. 1998.

[10] D. Maio and D. Maltoni. Real-time face location on gray-scale static images. *Pattern Recognition*, 33(9):1525–1539, Sept. 2000.

[11] H. Sawhney and S. Ayer. Compact representations of videos through dominant and multiple motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):814–830, Aug. 1996.

[12] M. Yang, D. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002.