Moving Objects Detection in Video Sequences Captured by a PTZ Camera

Li Lin, Bin Wang(✉), Fen Wu, and Fengyin Cao

School of Communication and Information Engineering, Shanghai University, Shanghai 200072, China
602552441@qq.com

Abstract. To solve the problem of detecting moving objects in video sequences which are captured by a Pan-Tilt-Zoom (PTZ) camera, a modified ViBe (Visual Background Extractor) algorithm, which is a pixel-based background modelling algorithm, is proposed in this paper. We divide a changing background scene into three parts. The first part is the new background region if a PTZ camera's field of view has been changed and we re-initialize background model of this part. The second is the disappeared area in the current frame and we decide to discard their models to save memory. Then the third part is the overlapping background region of consecutive frames. Via matching SURF feature points which are extracted only in background region we obtain an accurate homography matrix between consecutive frames. To ensure that the corresponding model from the former frame can be used in the current pixel, the homographic matrix should show a forward mapping relationship between the adjacent frames. Efficiency figures show that compared with origin ViBe algorithm and some other state-of-the-art background subtraction methods, our method is more affective for video sequences captured by a PTZ camera. More importantly, our method can be used in most of pixel-based background modelling algorithms to enhance their performance when dealing with videos captured by a moving camera.

Keywords: Detecting moving objects · PTZ camera · Background subtraction

1 Introduction

Moving objects detection is widely exploited as being the first layer of many computer vision applications, such as vehicle tracking [1], people counting [2] and many other related fields [3, 4]. In the last few years, various state-of-the-art background subtraction methods to detecting moving objects are proposed for video surveillance system with static camera. Simple moving objects detection algorithms regard a static frame as background reference. While finding an exact correct background reference is almost impossible due to the dynamic nature of real-world scenes. In order to adjust dynamic background and segment more accurate moving objects (foreground) from scenes, building a background model becomes the 'mainstream' approach. This is the basic

Y. Zhao et al. (Eds.): ICIG 2017, Part III, LNCS 10668, pp. 287–298, 2017.
https://doi.org/10.1007/978-3-319-71598-8_26

principle of background modelling: the current pixel or region compares with its background model, after that, unmatched areas will be labeled as foreground. Finally it will generate a binary mask to distinguish background and foreground.

Many well-known background modelling algorithms, like ViBe, Amber, SuBSENSE, etc. have achieved high-quality motion detection in video sequences captured by a stationary camera. However, when the stationary camera or PTZ cameras change their viewing area, these approaches are not suitable anymore.

Several difficulties in detecting motion based on PTZ cameras are listed as follows.

(a) *Motion Estimation Error of camera.* Motion between consecutive frames includes two independent parts: active motion of camera and motion of objects. Error is inevitable when estimating movement information of PTZ camera from video sequences. Such accumulative errors may have a badly influence on subsequent detection.
(b) *Multiresolution.* PTZ cameras have zoom-in zoom-out functions so that the same scene can be scanned by different resolutions. The background pixels undergoing these complex changes tend to be misclassified as foreground objects.
(c) *Real-time.* Many attempts have been accomplished to detect motion in a moving background by building a panoramic background image. Such background reference may perform well in a static scene because it can cover the whole area shoot by PTZ cameras. However, in order to store and make use of this large model, more memory and computational power will be required.

In this paper, we present a background modelling method to detect motion in video sequences which are captured by a PTZ camera. A basic background modelling algorithm, ViBe in [5], is employed to illustrate that our method can enhance performance of most pixel-based modelling algorithm when dealing with a moving scene. It changes the situation that most existing background modelling algorithms can not be applied to PTZ camera-based systems due to the presence of varying focal lengths and scene changes.

The remainder of this paper is organized as follows: In Sect. 2, we have a review on some typical background subtraction algorithms based on stationary cameras and PTZ cameras. The review introduces the main principle and relative merits of each algorithm briefly. Section 3 explains three key issues about background modelling and describes a modified ViBe algorithm in detail. Then we discuss experimental results and compare our results to other algorithms in Sect. 4. Section 5 concludes the paper.

2 Related Work

Over the recent years, numerous background modelling methods [5–9] have been developed. Most of these methods are just based on a static background. Gaussian Mixture Models (GMM) [6, 7] is widely used in real-life scenarios to handle a dynamic complex background (e.g. rain, swaying tree leaves, ripples). Non-parametric model based on Kernel Density Estimation (KDE) [8] also estimates background probability density functions. But differing from GMM, its background probability density functions

depend directly on the very recent observations at each pixel location. The feature is also important to building a background model. SuBSENSE (Self-Balanced SENsitivity SEgmenter) [9] proposed that individual pixels are characterized by spatiotemporal information based on color values and Local Binary Similarity Pattern (LBSP) features which describe local textures for background modelling.

To detect motion in video sequences captured by a PTZ camera, we should have knowledge about existing methods aim at a moving background. In general, the methods in the literatures of PTZ camera contain two main types: frame-to-frame (F2F) and frame-to-global. Frame-to-frame methods focus on the relationship between the consecutive frames. The current frame can reuse the information of overlapping regions from the previous frame. Kang et al. [11] present an adaptive background generation algorithm using a geometric transform-based mosaicking method. A homogeneous matrix, which describes a relation between adjacent images that have different pan and tilt angles, is used to project the existing background into the new image. This method, which differs from obtaining camera parameters by sensors directly, does not have to know the internal parameters of the PTZ camera. An algorithm proposed in [12] estimates parameters of the PTZ camera from meta data and frame-to-frame correspondences at different sampling rates. Besides using the low sampling frequency of meta data, two extended Kalman filters which uses the high frequency F2F correspondences are designed to enhance estimation accuracy. Beyond that, some methods are proposed to detect and track moving objects in a moving scene by applying of optical flow information [13]. Frame-to-global methods emphasize building and maintaining a panoramic background image of the whole monitored scene. Generating a panoramic background image based on image mosaic then finding the same scene in the panoramic background image by image registration, finally detecting moving objects by background subtraction is the most common approach. The problem of how to produce a background image is always discussed. The simplest case is to pan the camera 360-degree around its optical center, after that a panoramic mosaic image can be constructed on a cylindrical, squared, or spherical manifold [14]. In [15], the method extracts and tracks feature points throughout the whole video stream, and then make use of reliable background point trajectories to generate a background image. Sudipta N. Sinha et al. [16] describe a hierarchical approach for building multi-resolution panoramas by aligning hundreds of images captured within a 1–12× zoom range. In [17], a panoramic Gaussian mixture model (PGMM) covering the PTZ camera's field of view is generated off-line for later use in on-line foreground detection.

In this paper, we build a pixel-based background model with a frame-to-frame method. Compared with frame-to-global methods, our approach does not need the prior information or the off line computation process, which is hard to satisfy the requirement of real-time.

3 The Modified ViBe Algorithm

Each method has its own advantages and disadvantages. A background modelling algorithm should deal with at least three key issues. (1) *Initialization*. The initial process

determines elements in background models. Pixels are always characterized by color information. For more accurate results, texture information or mathematics operators can be added to background model, even if consuming more memory. Besides, the speed of initialization is another main factor to estimate algorithm performance. Methods, such as Gaussian mixture model and kernel density estimation, spend some time on training. But if a training process is set up in a model to detect motion in a moving background, it won't generate an appropriate model. Scenes captured by a PTZ camera are not fixed, so it is almost impossible to obtain enough static images for training. When it comes to the appearance or disappearance of background scenes, training becomes more difficult. Therefore, rapid and simple initialization process should be adopted to build a moving background model. (2) *Classification*. The similarity between new pixel and its background model decides whether the pixel belongs to background or foreground. In most cases, the decision threshold plays a key role in classification process. A high threshold causes false background pixels and many true background pixels will be omitted by a low threshold accordingly. Making thresholds adaptively is a good choice for different areas in a static scene. (3) *Update*. Changes in real-life scenes are inevitable. Each algorithm needs to choose its proper update policy to fit these changes. Which background model should be updated or how long can make one update? All kinds of background methods explain such problems in its update process.

In the following, we will describe the modified ViBe algorithm in detail according to the above three aspects. The overall flow of the proposed approach is illustrated in Fig. 1.

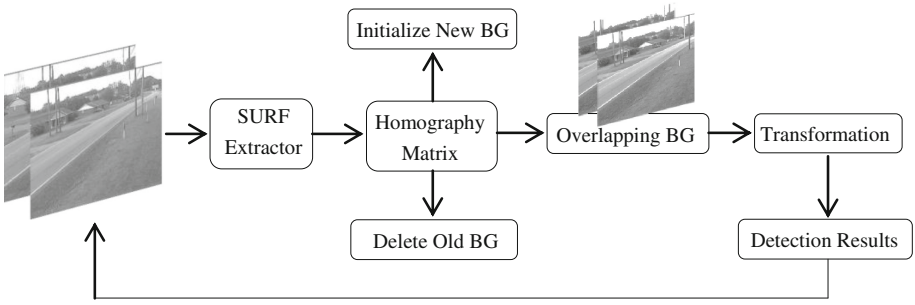


Fig. 1. Flow chart of the proposed approach. At first, matching the SURF feature points and computing the homography matrix between adjacent frames, then dividing the observed image into new background and overlapping background and doing corresponding managements, finally using the detection result which serves as feedback to delete the feature points in foreground when the next frame is captured.

3.1 Initialization

Compared with other classical algorithms, like Gaussian mixture model and kernel density estimation which initialized by some training frames, ViBe algorithm uses only one frame to initialize. Rapid and simple initialization is one of the remarkable advantages of ViBe algorithm.

First, a background model, noted by B, contains a set of N background samples for pixel located at x.

$$B(x) = \{v_1, v_2, \ldots, v_N\} \tag{1}$$

where v_i is the i^{th} background samples value. Due to every pixel value has a similar distribution with its adjacent pixels, ViBe algorithm fills background model $B(x)$ with eight neighborhoods pixel values of center pixel x. Usually the number of samples is $N = 20$. If size of video sequence is E × F, then the total size of background samples is E × F × N. It is random to select samples for a background model, so one of eight neighborhoods pixel value v_y of center pixel x may appear in $B(x)$ several times, or not even once.

When the perspective of PTZ camera changes, the adjacent frames can be divided into three parts. Distinctions between consecutive frames are not obvious, so we choose the images across twenty frames in Fig. 2. As is shown, region A is a new background scene. B is the overlapping region which appeared in the former image. And C represents a disappeared place. Obviously, the current image is composed of region A and B. In the same way, region B and C comprise the former image.

(a) (b)

Fig. 2. Three parts of (a) a latter frame and (b) a former frame: region A (new background), region B (overlapping background), region C (disappeared background).

Background models only in region A need to be reinitialized according to the initial approach above. It's unnecessary to preserve background samples in disappeared areas, so models in region C are directly abandoned to save memory. In region B, consecutive frames share a same background samples. The approach to apply the previous frame's background models to the current frame is described in Sect. 3.3. Therefore, initialization may operate in the whole modelling process, as long as background scene has spatial changes.

3.2 Classification

After initialization, we start to detect motion. Moving objects detection, regarding as a classification process, labels every pixel as one foreground pixel or a background pixel by a certain kind of rules. Then through the post-processing, a binary mask, where white

(pixel gray level = 255) represents foreground and black (pixel gray level = 0) represents background, is generated eventually.

If the distance between the pixel value v_x at location x and a given background sample value is smaller than the maximum distance threshold R, in other words, if inequality (2) is satisfied, we consider these two pixels are similar.

$$|v_x - v_i| < R \tag{2}$$

When a pixel finds th_{min} or more similar samples in its background model, the pixel will be classified as background.

$$NUM\{v_1, v_2, \ldots, v_N\} \begin{cases} \geq th_{min} & background \\ < th_{min} & foreground \end{cases} \tag{3}$$

where we fixed $R = 20, th_{min} = 2$. $NUM\{v_1, v_2, \ldots, v_N\}$ returns the number of similar samples in background model.

3.3 Update

Even if in static scene, the expectation, background without changes, almost never holds. Camera jitter, illumination or other background changes are unavoidable. But beyond that, translation, rotation, scaling of background scene captured by a PTZ camera lead to the most difficult problem. Reasonable update policy gives a hand to adapt such changes.

In our case, consecutive frames have overlap region B. The current pixel x located at (i, j) cannot use the previous model at the same location directly. We need to transform the former model into the current based on the following way.

The first major work figures out homography matrix between consecutive frames when the PTZ camera rotates around its optical center. We parameterize the pin angle of the PTZ camera by α, the tilt angle by β and the focal length by γ. $H_{n+1,n}$ presents a mapping relation from $(n+1)^{th}$ frame to n^{th} frame.

$$H_{n+1,\,n} = K_{n+1} R_{n+1} R_n^T K_n^{-1} \tag{4}$$

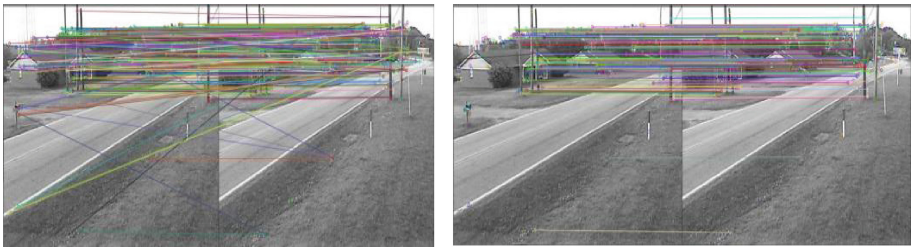
where

$$K_n = \begin{bmatrix} \gamma_n & 0 & 0 \\ 0 & \gamma_n & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

and

$$R_n = \begin{bmatrix} cos\alpha_n & 0 & sin\alpha_n \\ sin\alpha_n sin\beta_n & cos\beta_n & cos\alpha_n sin\beta_n \\ sin\alpha_n cos\beta_n & -sin\beta_n & cos\alpha_n cos\beta_n \end{bmatrix} \tag{6}$$

In our method, we use SURF (Speeded Up Robust Features) to represent the correspondences between two images with the same scene or object. Such a series of detectors and descriptors are achieved by relying on integral images for image convolutions detailed in [18]. SURF is scale and rotation invariant. It outperforms many previously proposed schemes with respect to repeatability, distinctiveness, and robustness, yet can be computed and compared much faster. We extract and match SURF descriptors in two adjacent images. Note that these feature points are located only in background according to the classification results of the previous frame. When matches located in moving objects occur, homography matrix, which is computed through the correspondence relationship among matching points, can't express background transformation relation precisely. Shown in Fig. 3, Outliers are filtered by RANSAC algorithm, which achieves goals via repeatedly selecting a set of random date subset.



(a) *(b)*

Fig. 3. RANSAC algorithm removes some mismatching points. (a) Original matching images. (b) Matching images after filtering.

To enable every current pixel in overlap region have a certain model from history, Homography matrix, noted H, indicates the mapping from the current image to the former image. As shown in Fig. 4, the previous location (i'_{t-1}, j'_{t-1}) of pixel x which located at (i_t, j_t) may be a non-integral type. Thus we use bilinear interpolation to select background sample values from previous models for x. Formulization (7) explain the way to figure out one of the background sample values. At last background model at x is updated through calculating bilinear interpolation for N times.

$$v_t^x = mnv_{t-1}^a + m(1-n)v_{t-1}^b + (1-m)(1-n)v_{t-1}^c + (1-m)nv_{t-1}^d \tag{7}$$

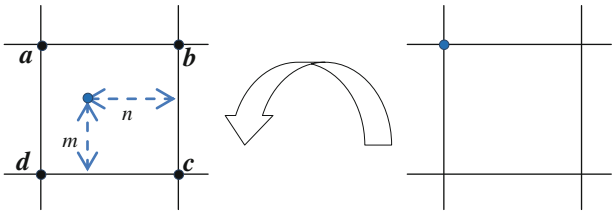


Fig. 4. The previous pixel and the current pixel based on forward mapping

Meanwhile, we also need to update background model by inserting the current pixel x. Our method incorporates three of important components from ViBe algorithm: a conservative update policy, a random update policy and spatial propagation of background samples.

A conservative update policy considers a pixel, only if it has been classified as background, qualified to update its background model. It means samples in foreground are never included in background model. Conversely, there is a blind update policy using not only background but also foreground to update models. The principal shortcoming of blind update is that it may lead to more false background and poor detection of slow moving objects.

Many background modelling methods use first-in first-out policy to update. It holds that the recent background sample has more efficacies but the oldest does not. In spite of ignoring the importance of temporal relationship, updating background samples randomly is still simple but effective in our methods. Observation classified as background will replace one of its background samples. The replaced sample is selected randomly. In other words, the probability of every sample being abandoned is 1/N. Considering together with spatial propagation of background samples, a sample in the model of a pixel y, which is one of eight connected neighborhood of x, is also replaced by the current observation. Such update policy takes into account spatial relationships among incoming pixel with its surrounding.

The most difference of our method from original ViBe algorithm is the model update rate. ViBe algorithm sets its time subsampling factor as 16. But in terms of detection in moving background scenes, it is necessary to update each background pixel model more quickly. So that we update background model of pixel x for every new frame as long as x is classified as background.

4 Experiments

This section reports the performance of our approach with the experiment results on the PTZ video sequences from the public Change Detection benchmark 2014. The percentage of correct classification (PCC), which wants to be as high as possible, is used to evaluate our approach.

$$PCC = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

where TP (True Positives) counts the number of correctly pixels classified as foreground, TN (True Negatives) counts the number of correctly pixels classified as background, FP means the number of background pixels incorrectly classified as foreground and FN accounts for the number of foreground pixels incorrectly classified as background.

From detailed discussion from ViBe algorithm in [5], we fix the radius $R = 20$ and the decision threshold $th_{min} = 2$. The only difference of our method from original parameters is the time sampling factor T, which set as 16 formerly. Detection results and PCCs for model time subsampling factor T ranged 1 to 7 are displayed in Fig. 5. Obviously the best results are obtained for $T = 1$. It seems that a smaller time subsampling factor,

which indicates a faster update of background model, may implement a more accurate result for moving background scenes.

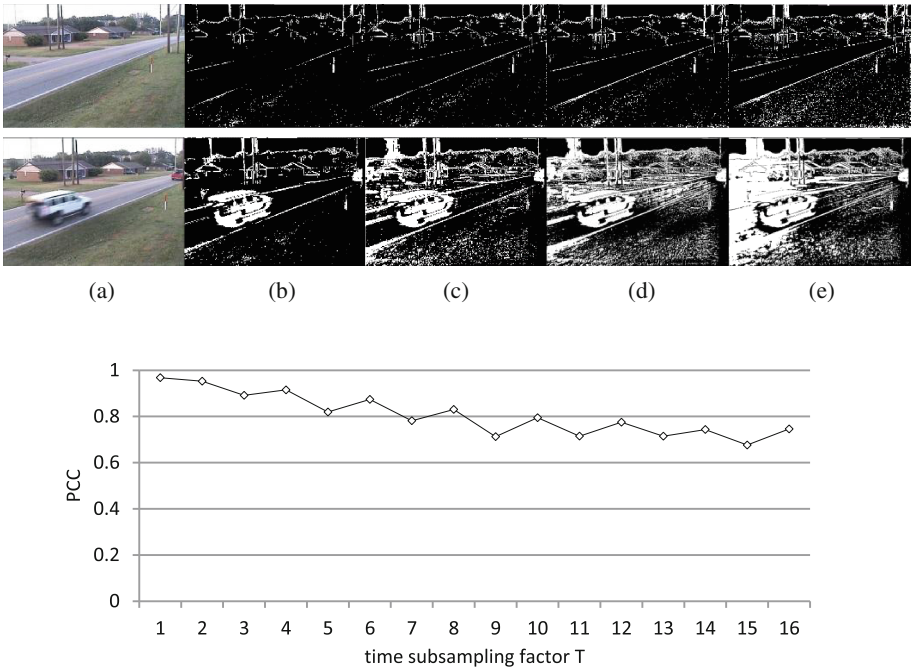


Fig. 5. Detection results and PCCs for time subsampling factor T ranged 1 to 7. (a) Input image. (b) $T = 1$. (c) $T = 3$. (d) $T = 5$. (e) $T = 7$. (f) PCCs.

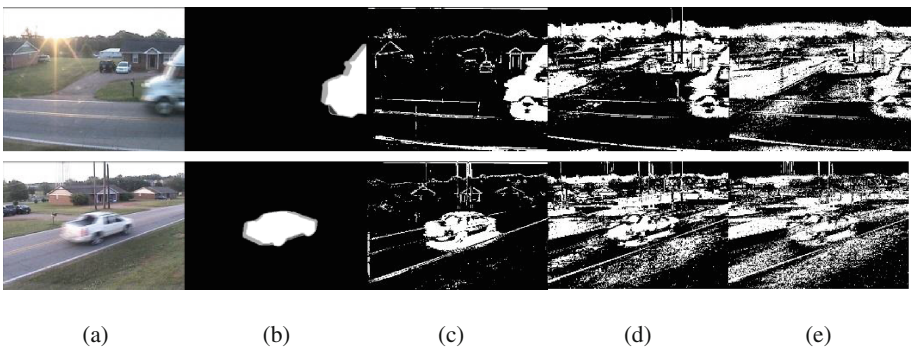


Fig. 6. Comparative pure segmentation results of three background modelling techniques for continuous pan video sequence. (a) Input images. (b) Ground truth. (c) Our methods results. (d) Original ViBe algorithm. (e) Gaussian Mixture Models

Figure 6 shows input images, ground truth and pure detection result without any morphological operations or noise filtering. Edges as shown are almost eliminated by a post-process. Visually, the result of our methods is better. Background modelling algorithms, like GMM, are not proper anymore. Yet every method, so does ours, have a problem that foreground pixels will be initialized into background model when there are any moving objects in new background, furthermore such error lasts for a long time. Just in the same way to remove ghosts, if the current scene not going away immediately, this mistake will be resolved by spatial propagation of background samples in the end.

We combine our idea with ViBe algorithm to illustrate that the performance after our processing is improved when dealing with videos captured by a moving camera. More importantly, our method can be used in most of pixel-based background modelling algorithms to enhance their performance. To compare our methods in handing such a moving background mathematically with original ViBe algorithm and other several methods, other metrics proposed in the change detection website are also considered here. F-Measure, which is the weighted harmonic mean of 'precision' and 'recall', indicates overall performance well. The 'precision' is the ratio between the number of correctly classified as foreground and the pixels which are classified as foreground regardless of the correct. The 'recall' is used to describe the accuracy of whether the true foreground pixels are correctly classified or not. So we use F-Measure to obtain a single measure to evaluate different methods then rank them.

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

$$F - Measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{11}$$

From the experimental result shown in Table 1, it can be clearly seen that our approach after sample post-process achieves much better performance than original ViBe algorithm and other pixel-based algorithms in detecting continuous panning background. This indicates that our method is extremely beneficial to the original background modelling algorithm to adapt the difficult moving scenarios captured by a PTZ camera.

Table 1. Average performance comparison of different models.

	PCC	Precision	Recall	F Measure
Ours	0.9679	0.1324	0.7424	0.2247
ViBe	0.7822	0.0148	0.5135	0.0288
GMM	0.7929	0.0139	0.4578	0.0270
KDE	0.7516	0.0181	0.7261	0.0353

5 Conclusion

Over the recent years, numerous background modelling methods have been developed. However, most existing work proposed for fixed cameras can not be directly applied to PTZ camera-based systems due to the presence of varying focal lengths and scene changes. Furthermore, there is much less research work for PTZ camera-based background modelling. Most methods generate a background mosaic image then use the simplest background difference method to obtain a binary mask. In this paper, we have presented a modified Vibe algorithm to detecting moving objects in video sequences which are captured by a PTZ camera. More importantly, our method can be used in most of background modelling algorithms to suit a moving scene. We tested the performance of the method in comparison with classical existing methods. It outperforms these methods in motion detection when the background scene keeps moving.

As for future extension, we are trying to combine our method to other more complex pixel-based background modelling algorithms. In addition, a detailed analysis of different application with respect to a faster moving scene, which may shoot by car cameras or unmanned aerial vehicles, is also our future consideration.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (grant number: 61601280) and Key Laboratory for Advanced Display and System Applications (Shanghai University), Ministry of Education of China (grant number: P201606).

References

1. Lipton, A.J., Fujiyoshi, H., Patil, R.S.: Moving target classification and tracking from real-time video. In: IEEE Workshop on Applications of Computer Vision. IEEE Computer Society, p. 8 (1998)
2. Hou, Y.L., Pang, G.K.H.: People counting and human detection in a challenging situation. IEEE Trans. Syst. Man Cybern. Part A Syst. Hum. **41**(1), 24–33 (2011)
3. Cutler, R., Davis, L.: Real-time periodic motion detection, analysis, and applications. IEEE Trans. Pattern Anal. Mach. Intell. **22**(8), 781–796 (1999)
4. Wren, C.R., Azarbayejani, A., Darrell, T., et al.: Pfinder: real-time tracking of the human body. In: International Conference on Automatic Face and Gesture Recognition, pp. 51–56. IEEE Xplore (1996)
5. Barnich, O., Van, D.M.: ViBe: a universal background subtraction algorithm for video sequences. IEEE Trans. Image Process. Publ. IEEE Signal Process. Soc. **20**(6), 1709–1724 (2011)
6. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In: 1999. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, p. 252. IEEE Xplore (1999)
7. Zivkovic, Z.: Improved adaptive gaussian mixture model for background subtraction, vol. 2, pp. 28–31 (2004)
8. Elgammal, A., Harwood, D., Davis, L.: Non-parametric model for background subtraction. In: Vernon, D. (ed.) ECCV 2000. LNCS, vol. 1843, pp. 751–767. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-45053-X_48

9. Wang, B., Dudek, P.: AMBER: adapting multi-resolution background extractor. In: IEEE International Conference on Image Processing, pp. 3417–3421. IEEE (2014)
10. St-Charles, P.L., Bilodeau, G.A., Bergevin, R.: SuBSENSE: a universal change detection method with local adaptive sensitivity. IEEE Trans. Image Process. **24**(1), 359–373 (2014)
11. Kang. S., Paik, J.K., Koschan, A., et al.: Real-time video tracking using PTZ cameras. In: Proceedings of SPIE - The International Society for Optical Engineering, vol. 5132, pp. 103–111 (2003)
12. Wu, S., Zhao, T., Broaddus, C., et al.: Robust pan, tilt and zoom estimation for PTZ camera by using meta data and/or frame-to-frame correspondences. In: International Conference on Control, Automation, Robotics and Vision, pp. 1–7. IEEE (2007)
13. Doyle, D.D., Jennings, A.L., Black, J.T.: Optical flow background subtraction for real-time PTZ camera object tracking. In: Instrumentation and Measurement Technology Conference, pp. 866–871. IEEE (2013)
14. Mann, S., Picard, R.W.: Virtual bellows: constructing high quality stills from video. In: Image Processing, 1994, Proceedings, ICIP-94, IEEE International Conference, vol. 1, pp. 363–367. IEEE (2002)
15. Jota, K., Tsubouchi, T., Sugaya, Y., et al.: Extracting moving objects from a moving camera video sequence. IPSJ SIG Notes CVIM **2004**, 41–48 (2004)
16. Sinha, S.N., Pollefeys, M.: Pan–tilt–zoom camera calibration and high-resolution mosaic generation. Comput. Vis. Image Underst. **103**(3), 170–183 (2006)
17. Xue, K., Liu, Y., Ogunmakin, G., et al.: Panoramic gaussian mixture model and large-scale range background substraction method for PTZ camera-based surveillance systems. Mach. Vis. Appl. **24**(3), 477–492 (2013)
18. Bay, H., Tuytelaars, T., Gool, L.V.: SURF: speeded up robust features. Comput. Vis. Image Underst. **110**(3), 404–417 (2006)