International Journal on Artificial Intelligence Tools
Vol. 27, No. 2 (2018) 1830001 (25 pages)

DOI: 10.1142/S0218213018300016

World Scientific
www.worldscientific.com

Features-Based Moving Objects Tracking for Smart Video Surveillances: A Review

Nor Nadirah Abdul Aziz*, Yasir Mohd Mustafah†, Amelia Wong Azman‡,
Amir Akramin Shafie§, Muhammad Izad Yusoff¶, Nor Afiqah Zainuddin‖ and
Mohammad Ariff Rashidan**

Department of Mechatronics Engineering
International Islamic University Malaysia, Kuala Lumpur, Malaysia
**nornadirahaziz89@gmail.com*
†yasir@iium.edu.my
‡amy@iium.edu.my
§aashafie@gmail.com
¶mizadyusoff@gmail.com
‖fiqahzainuddin@gmail.com
***ariffrashidan@gmail.com*

Received 19 March 2014
Accepted 16 October 2017
Published 29 March 2018

Video surveillance is one of the most active research topics in the computer vision due to the increasing need for security. Although surveillance systems are getting cheaper, the cost of having human operators to monitor the video feed can be very expensive and inefficient. To overcome this problem, the automated visual surveillance system can be used to detect any suspicious activities that require immediate action. The framework of a video surveillance system encompasses a large scope in machine vision, they are background modelling, object detection, moving objects classification, tracking, motion analysis, and require fusion of information from the camera networks. This paper reviews recent techniques used by researchers for detection of moving object detection and tracking in order to solve many surveillance problems. The features and algorithms used for modelling the object appearance and tracking multiple objects in outdoor and indoor environment are also reviewed in this paper. This paper summarizes the recent works done by previous researchers in moving objects tracking for single camera view and multiple cameras views. Nevertheless, despite of the recent progress in surveillance technologies, there still are challenges that need to be solved before the system can come out with a reliable automated video surveillance.

Keywords: Video surveillance system; object detection; visual tracking; modelling; features.

1. Introduction

The Closed-Circuit Television (CCTV) camera is an important tool used mainly in two applications; (1) control and regulation purposes and (2) security purposes. For control and regulation purposes, CCTVs are used to enforce traffic regulations,[1] production control and quality assurance.[2] Meanwhile, CCTVs camera in security purposes are used as verification of alarms i.e., whether there is an intrusion[3] or fire occurrence,[4] detection of criminal offences,[5] as well as documentation of money transfer at an ATM[6] to name a few.

Video surveillance systems may consist of a single camera or multiple cameras and the coverage of the surveillance systems are defined by the field of view of the cameras. Recent works on tracking moving objects using multiple cameras is increasingly popular. The multi-cameras environment covers more physical space than a single camera view which provides a more comprehensive view about the crime scene. The coverage of multi-cameras surveillance is defined by the cameras' field of view (FOV) which is either overlap or non-overlap depending on the direction of the camera being installed. In many premises, human operator is assigned to monitor the video continuously to detect any suspicious activity. There are several limitations in the conventional video surveillance system with the high cost of assigning the human operator as the main issue.[7] Secondly, the conventional system supplies only a linear access to the video data.[8] An automated video surveillance system is essential to ease the burden and increase the efficiency of human operator.

Automated video surveillance system intends to exploit the videos captured from the CCTVs using software to automatically identify the objects. The automated system involves processing methods on the visual images such as computer vision techniques that give output without human interruption. Computer vision techniques include method of acquiring, processing and analyzing visual images from a real world and converting the raw data into a form that can be processed by the machine. Applications for surveillance system in snatch theft crime detection cover a large scope in machine vision, including detection and tracking of the object that may be related to the crime itself. The aim of machine vision research is to provide computers with humanlike perception ability. The application has the ability to detect the moving objects that can reduce the amount of data needed for processing and applicable for real-time performance. Meanwhile, tracking of moving objects can give a unique identity to each of the object detected in the video consistently. From the tracking results, it can provide important information about the objects such as the object's appearance, and biometric information including the object's height. Depending on the types of application, the object of interest in a research might be differed, e.g., recognizing the plate numbers of vehicle[9] or detection of crime offences based on the unusual behavior of human.[10]

In general, the framework of a multi-camera surveillance system encompasses a large scope in machine vision that includes background modeling, objects detection, moving objects tracking, and requires fusion of information from the camera networks. The foremost task in many surveillance applications is to detect the objects of interest as they appear within the camera view. The detection of moving objects is important for target tracking. Object detection is a challenging task especially for distributed cameras as the objects belonging to the same class, such as human, might significantly differ in appearance due to clothing, illumination, pose and camera parameters factor. When the moving object is successfully detected in the video surveillance system, the next step is to track the objects within the disjoint views. It is important to accurately track the moving objects across the different views as it can help the enforcers to get the visual tag of the criminals from various sources and track them within a short concentration span. This is done so as to eliminate disregarding important crime details. However, tracking moving objects in multi-cameras environments is more challenging compared to single camera view. This is due to different illumination conditions, viewing angles and poses between different camera views. Besides, there is no spatial continuity between cameras that have non-overlapping views. Figure 1 shows the flowchart of a general framework for tracking across multiple camera views.

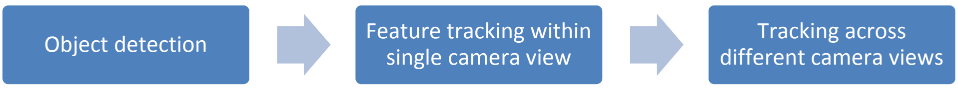

Fig. 1. General framework for tracking across multiple camera views.

2. Object Detection

Object detection is the first step before extracting any information for any surveillance applications such as recognition and tracking.[11] Generally, object detection comprises of several sequential processes: (1) Pre-processing, (2) Object segmentation, (3) Mathematical morphology and (4) Blob extraction.

2.1. *Pre-processing*

Operation on images at the lowest level of abstraction with both input and output are intensity images, are called pre-processing. Pre-processing is applied to suppress noise or enhance some image features that are important for further processing.[12] In Ref. 13, pre-processing is used on the acquired images to minimize the effect of varying illumination and noise. The varying illumination is minimized by using illuminate function which enhance the image contrast. For noise reduction, the image frame is smoothened by Gaussian filter.[13,14] In fact, this process also helps in reducing aliasing in the case of downsizing of high resolution videos when necessary.

2.2. *Object segmentation*

In visual surveillance, object segmentation is done by assigning a label to each pixel so that the pixels with identical label will share certain characteristics. As a result, object segmentation gives a set of segments from the image that belongs to a moving object. Generally, there are two approaches commonly used by earlier researchers that are (1) background subtraction and (2) optical flow.

2.2.1. *Background subtraction*

Background subtraction is a process of finding deviation between background image and foreground image. Image binary pixels that deviated from the background pixels would be the object of interest. Background subtraction method is relatively less time consuming.[15] In Ref. 16, a method of using a robust light-weight salient foreground detection algorithm is proposed to implement the background subtraction.

Background subtraction that is based on color information is susceptible to sudden changes in illumination. In contrast to gradients of image, they are less sensitive to the changes in illumination.[17] In Ref. 18, a new advance background differencing that combined color and gradient information is proposed to perform quasi illumination invariant background subtraction. It is based on multi-level stages. This technique solve the problems of drastic illumination changes due to adverse weather conditions, repositioning of static background objects and initialization of background model with moving objects present in the scene case.

In Refs. 19 and 20, they utilized a pixel-wise median filter method to built the background model under assumption that a moving object will not stay at the same position for more than certain period. It is based on sliding-window spatial filter that replaces the center value in the window with the median of all the pixel values in window. A median filter can build the background even when the moving objects exit in the scene. Median filter can remove an impulsive noise, but, it usually requires a large amount of memory to save the number of frames at a time. The background subtraction is done based on color and edge density.[19]

Zhang and Ding[21] proposed an adaptive algorithm that is based on Local Binary Pattern (LBP) texture distributions. This technique can tolerate considerable illumination variations in common natural scenes. This is because the LBP features are invariant to monotonic gray-scale changes. Additionally, this technique is computationally fast that is suitable for real-processing.

In Ref. 22, they presented a new method inspired by the Gaussian mixture model technique based on YCbCr color space. The background pixel value is computed based on the mean value of all similar pixels within a certain period, given that the stability value of each pixel is equal to the desired number of frames. The background information is updated according to three conditions: (1) pure background, (2) illumination change and (3) static object. Nevertheless, the contour of the segmented region is usually broken due to imperfect background modelling.[21]

Some background subtraction techniques have problems especially for outdoor environment, e.g., the changes in illumination when the sunshine is covered by the clouds. In Ref. 23, they proposed a method by modelling each background pixel using M Gaussian distribution. It is one of the successful solutions to this problem that used multicolor background model per pixel. This method also has been adapted by Refs. 24–29 to detect foreground regions. This method also has been proven to solve other problems such as repetitive motion from clutter and long-term scene changes. However, the computation cost for the M Gaussian distribution is too high and thus, reduces the system efficiency.[21] Nevertheless, in Ref. 30, they emphasized that the computational of the M Gaussian distribution will become larger if the number of Gaussian is greater. The optimum value for number of Gaussians is between three to seven, thus, in Ref. 30, they have selected four Gaussians for real-time processing as described in their paper.

In Ref. 31, they proposed an adaptive background reconstruction based on pixel intensity classification and mode filtering. The intensity value of image pixel is compared between current frame and reference frame based on several image patches. The background is reconstructed if the intensity values greater than a predefined threshold. Finally, mode filtering is used to obtain the background model. This technique can adapt with adverse weather condition and does not require any non-moving object in background reconstruction.

Shafie *et al.*[32] improved method by Ref. 33 by comparing the luminance value of all pixels between current frame and reference frame. The current frame is adopted as the background image if the value remains unchanged. Shadow can affect the performance of the foreground detection, thus, in Ref. 32, they adopted method based on color information, i.e., chromaticity and luminance[29–34] to remove the shadow. Background subtraction is used to obtain the foreground object.[32,35]

In Refs. 36 and 37, they proposed a block based background subtraction approach for high resolution images compared to previous method using pixel based approach in Ref. 38. In Refs. 36 and 37, the authors exploited spatial co-occurrence of image variations by using the average intensity value of R, G and B color components in $N \times N$ block. This method is unsusceptible to noise or small changes in background, and require less memory space and involves simple arithmetic operations, thus, applicable in real-time application. However, for this method, the foreground can only be extracted for coarse-resolution level.

2.2.2. *Optical flow*

Optical flow uses motion vectors from the objects over time to detect a moving region.[39] It is a complex method and is not reliable for a real-time processing.[15] This technique measures the optical flow based on inter-frame difference method.

In Ref. 13, they proposed to use optical flow to extract the information about the movement between two consecutive image frames. The segmented regions are only used for matching task, thus reducing the matching complexity. Method proposed by Brox *et al.*[40] is applied to compute the optical flow. This method is robust under a considerable amount of noise. Although a lot of enhancements have been done by Brox's technique compared to the traditional method that is through Horn Schunk technique the traditional method is more preferable due to the global smoothness constraints that results in smoother optical flow vector with low execution time.[41]

In Ref. 42, they proposed a method for background modelling based on fusion of optical flow and color information. However, object segmentation based on optical flow is computationally expensive and not reliable for real-time processing. Using optical flow in object segmentation is also not suitable for objects that have very inhomogeneous and non rigid movements.[43] In Ref. 39, they further emphasize that optical flow would change dramatically in highly textured regions, around moving boundaries and depth discontinuities.

2.3. *Mathematical morphology*

A binary image that contains discrete noises can be smoothened by using morphological operation, which relies on relative ordering of pixel values. This technique probe an image with a small shape called structuring element. In Refs. 15, 29, 32 and 36, they utilized morphological operation on the threshold image obtained from optical flow technique. In Refs. 21–22 and 44–45, they used opening and closing operations to make up the fragment effect of the segmented foreground region.

2.4. *Blob extraction*

After obtaining the segmented region, the blob extraction is done by applying Component Connected Labeling (CCL) algorithm.[21,46–48,66] It is used to obtain the connected regions in binary images.[11] This method groups the binary image based on the nearest neighborhood binary pixels.

3. Visual Tracking

Visual tracking can be categorized into two parts[49] that are: (1) Modelling and (2) Tracking. The appearance model gives *a priori* information about the object and can be updated for each new frame. The appearance can be modelled using shape, templates, histograms or parametric representations of distributions. The second step in visual tracking is how to use the model to find the location of the object in the next frame. One of the simplest ways is by correlating the new frame with the appearance model and finding the maximum response. Other method that has become a popular trend is by applying tracking by detection. This method uses machine learning language technique to train a discriminative classifier for the object appearance. It is done by classifying image regions to find the most possible location for the object in the next frame.

3.1. *Types of features used for modelling*

Recently, most of the surveillance systems rely on multiple features to model the object's appearance. However, there are some of methods that also focus on using single feature for tracking. Here, types of feature used for modelling can be categorized into two categories: (1) single feature and (2) multiple features. Generally, the method used in single camera can be easily further extended for multiple cameras. Here, it is assumed that the system has been performed for single camera, thus, the analysis only focuses on features used for multiple cameras.

3.1.1. *Single feature*

Color provides a superior performance for visual tracking. The choice of color feature is important to ensure that the tracker can track the objects successfully. In Ref. 50, they emphasized that color attributes or Color Names (CN) has been proven to give excellent results for object recognition recently.

Danelljan *et al.*[51] proposed an adaptive low-dimensional variant of color attributes for real-time visual tracking. They extended Circulant Structure of Kernels (CSK) tracker by Ref. 50 with color feature which gives an excellent result for object recognition due to good balance between photo metric invariance and discriminative power. They used mapping provided by Ref. 52 to associate RGB observations with 11 dimensional color representations. The normalization of color is done by projecting CN to an orthonormal basis of 10-dimensional subspace to the center the CN. They applied dimensionality reduction technique to improve the computational speed. This method is reliable for real-time performance with the speed more than 100 frames per second (fps). However, it is not reliable for several attributes, i.e., scale variation, deformation and fast motion.

In Ref. 24, they represented object appearance using several regular patches based on dominant colors. The appearance model is obtained by accumulating those color patches which is repeatedly occur over the sequence. This method is robust to illumination changes and partial occlusion. However, it is not reliable for tracking people with similar appearances and under illumination variation.

Huang *et al.*[53] adopted method similar to Ref. 54 to divide the region into three main parts based on Fig. 2(b). The general chromatic content is obtained using HSV histogram. The pre-region color displacement is extracted using Maximally Stable Color Regions (MSCR) based on Fig. 2(c). To minimize the pose variation effects, the extracted features are weighted by distance vertically. The patches are sampled at different scales to solve the issue of scale variation. Dense sampling strategy and combination of color content are implemented to characterize the patch content as illustrated in Fig. 2(d). All these features are converted into BoF vectors and then clustered using Hierarchical k-means (HKM) for model vocabulary building. For the object matching, a linear kernel supported vector machines (SVMs) is utilized to classify the BoF vectors. This method is computationally fast due to limited sampling patches.

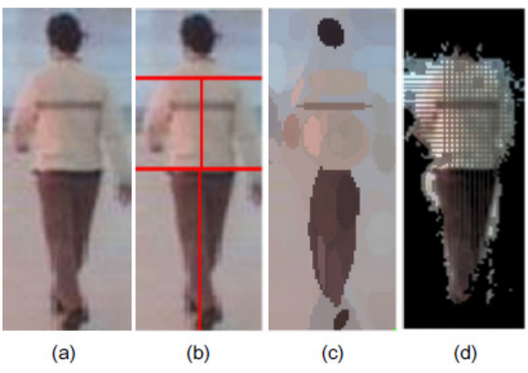

Fig. 2. (a) Image of a person; (b) The horizontal and vertical lines split the image into three main parts and pose variation estimation; (c) Maximally Stable Color Regions (MSCR); (d) Sampling patches within silhouette.[53]

3.1.2. *Multiple features*

Homography transformation can be used to solve the issue of consistent labeling.[21] The authors utilized color, gait period and spatial features to compute the similarity of each feature between the cameras. Under non-overlapping views, the object's position is predicted based on object average movement vector.

Spatio-temporal cues in a single camera tracking is used to estimate the directional angle of each object based on the pedestrian movement sequences that can deal with changes in viewpoint.[54] Color feature from patches of three region, i.e., head, upper and lower body, are extracted using K-mean clustering.[22] A stochastic matching is used to find similarity between reference and candidate blobs and it can be used to compensate small variances in poses.[54]

Combination of features, i.e., side of entry, color histogram, height, moving direction, speed and travel time are used to match objects across disjoint views.[16] Each candidate object, $i \in \{1 \ldots N\}$, has K different features and for each feature, $j \in \{1 \ldots K\}$, a similarity score s_j^i is calculated and given a weight w_j. The overall similarity score is computed based on Eq. (1).

$$O = arg \max_{i \in N} \sum_{j \in K} \left(w_j s_j^i \right) \tag{1}$$

Multiple features combined with empirically determined weights might be not so effective and resulted to some error.[16] Direct comparison between features from corresponding regions may cover different area in real world.[52]

In Ref. 55, they built links between cameras based on region mapping matrix and region matching weight to solve this problem. The path that link between the exit/entry zones of directly connected cameras is represented by camera link model. Each of exit/entry observation stores temporal, color and texture features of a person that has entered or left the camera FOV. The camera link model only can be updated when the matching score in Eq. (2) is higher than a certain threshold, i.e., α_i is the weight for the feature distance i.

$$score = -\sum_{i=1}^{N_{feature}} \alpha_i \times feature_{dist_i} \tag{2}$$

Xu *et al.*,[56] on the other hand, they proposed a method based on global graph to model the relationships between all local camera observations. They proposed to use speed and direction of motion that are less sensitive to illumination changes, variation of poses and occlusion variations. The nodes send these cues together with the object's spatial and temporal information to a central node. A graph is built at the central node to model the relationships between all the observations.

In Ref. 26, they used fusion of shape, color, texture and biometric information to measure the similarity between two observations. The shape is computed by using ratio of width over height or ratio of area over perimeter. The combined similarity measures are computed as in Eq. (3).

$$s_{comb}(\tau_a, \tau_b) = median\{s'_k(\tau_a, \tau_b)\} \tag{3}$$

where s'_k is the normalized similarity matrix, τ_a and τ_b are the points in track a and b from two observations. The matching score are computed based on Eq. (4).

$$m_\theta = \max_{j:l_j=\theta} s_k(\tau_i, \tau_j) \tag{4}$$

Mehmood and Khawaja[13] proposed an appearance based human matching for multiple non-overlapping cameras FOV. Lighting variations that affect intensity features are compensated using region based shape description and all the cameras must view the same ground plane.

If a target leaves the camera's field of view, it is expected to appear in the neighboring camera in a short time. Thus, in Ref. 46, they used time delay to track the objects across multiple camera views. The objects appearance are built based on HSV histogram and compared using Bhattacharyya distance. Unfortunately, this method is not reliable under different pose and unsuitable for multiple objects tracking.

Table 1. Features with their advantages and limitations.

Features	Advantages	Limitations	Ref.
Color	• Insensitive to non-rigid deformation • Insensitive to partial occlusion • Insensitive to target rotation • Insensitive to overlap	• Sensitive to illumination • Suffers from presence of confusing colors in the background	[45, 57–59]
Texture	Robust to brightness change or illumination variation	Requires a processing step to generate texture descriptors	[55, 57, 60]
Height	Insensitive to illumination	Robust to partial occlusion	[26]
Shape	• Insensitive to illumination • Less consistent within frames	Robust to partial occlusion	[26, 61]
Edge	• Insensitive to illumination • More robust to the presence of confusing colors in the background	Sensitive to clutter	[58]
Speed/ motion	• Insensitive to illumination • Less consistent within frames	Less consistent within frames. In real life, the object movement is inconsistent.	[61]
Travel time	Insensitive to illumination	Less consistent within frames.	[46, 61]

3.2. *Related methods used for modelling*

Model is representation of color, edge, texture, or points of an object. Below is the list of common approaches used by previous researchers to represent the object model.

3.2.1. *Histogram*

In visual surveillance, histogram represents distribution of extracted features in an image. Histograms are proven a powerful representation for image data. Histograms have been used widely due to its robustness to changing of object poses and shapes.[62–64]

3.2.1.1. Color histogram

Color histogram is used to model the object appearance within the image. It is reliable when the object's orientation changes.[65] The color feature is extracted from the image is divided into equal number of bins. In Refs. 57 and 66, they represented each bin in the color histogram to Red, Green, Blue (RGB) range. However, color histogram based on similarity measure is lack of information about the object spatial layout, thus, not discriminative enough.[67]

3.2.1.2. Edge Orientation Histogram (EOH)

Liu and Zhang[65] emphasized that edge orientation histogram (EOH) is more reliable under illumination changes. EOH has been used to identify the objects based on the distribution of edge point orientation of the object. Canny detector is the most powerful edge detector used to obtain the edge structure in the moving object region. It includes the weak edges in the output if they are connected to strong edges by incorporating the weak pixels that are 8-connected to the strong pixels.[68] However, in Ref. 45, they applied Sobel operator to detect the edges. In Ref. 64, they demonstrated the characteristic of EOH for pedestrians. Based on Fig. 3, most of the edge points consist of vertical edges.

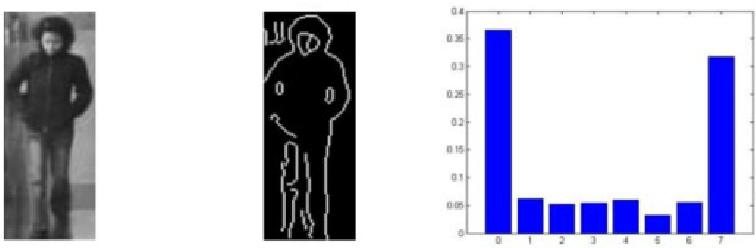

Fig. 3. Edge orientation histograms for pedestrian.[65]

The advantages of using EOH are: (1) it reflects the structures of the objects, (2) robust against region color changes, (3) robust to partial occlusion, background clutters and appearance changes, (4) 2D-scale and 2D-translation invariant and (5) it is robust when the background has similar colors with the foreground.[65]

3.2.1.3. Local Binary Pattern (LBP) histogram

Local Binary Pattern (LBP) is used to extract the texture feature of the moving objects across the frames.[28,57] LBP is fast due to its computational simplicity thus, make it suitable for real-time operation. It can be extracted faster in a single scan through the raw image with a low-resolution dimensional space. This is important for real-life applications when only low-resolution video is available. LBP features are robust to illumination changes due to its gray-scale invariance characteristic and deformation.[23,69]

3.2.1.4. Histogram of Oriented Gradient (HOG)

Dalal and Triggs[70] proposed a new method for feature extraction based on oriented gradient or also known as Histogram of Oriented Gradient (HOG). It is a shape descriptor that can distinguish humans from the background and robust to illumination variations.[25]

3.2.2. *Parametric representation of distributions*

Parametric representation of distributions express the points in a distribution as a functions of variable called parameter, e.g., K-mean clustering. Here, the *k* points of K-mean represent a parametric distribution.

3.2.2.1. K-mean clustering

Lin and Huang[22] utilized color clustering method based on K-mean clustering method as shown in Fig. 4. In Ref. 71, they employed K-Means clustering to determine non-uniform color histogram bins. The cluster is a non-uniform rectangle. For adjacent rectangle that has an overlapping region, the pixels are clustered based on minimum distance from cluster centers. In Ref. 72, they utilized K-mean algorithm to extract the most prominent RGB color of the object. K-Mean clustering is simple and flexible,[73] but, the performance depends on initial centroids, thus it does not guarantee for optimal solution.

(a) (b)

Fig. 4. Color clustering: (a) Large object detection; (b) Color clustering of (a).[22]

3.3. *Related algorithms used for tracking*

3.3.1. *Kalman Filter*

In Ref. 24, they proposed to use Kalman Filter to track the position and moving vector of the pedestrians. Meanwhile, Bilal *et al.*,[74] adopted Kalman Filter for face and hand tracking. This method utilized the state model to predict the new state of motion status and in Ref. 74, the prediction of hand blobs would be useful for sign language recognition. Prediction may reduce the computational cost because if the predictor tells the position of an object, only certain parts of the image must be processed instead of considering the whole frame. Kalman Filter works based on single hypothesis such that an observation is associated to only one of the existing tracks.[75] This method is not reliable when the number of object increases. Thus, Bazzani *et al.*[75] extended Kalman Filter method based on Multi-Hypothesis Kalman Filter (MHKF) to solve this problem. However, this method have some drawbacks i.e., (1) Targets are tracked with multiple tracks, leading to a abundance of tracks, (2) The target ID changes after an occlusion and (3) Tracking fails when the object's motion is non-linear.

3.3.2. *Particle filter*

Particle filter is a statistical filtering method based on Monte Carlo recursive Bayesian estimation. It is a sampling method for approximating a distribution that evolves in time.[76] Particle filter is very suitable for tracking multiple objects.[77] Nevertheless, the computational is complex since all the particles needed to achieve required approximation accuracy.[77–79]

Rahimi *et al.*[25] used weighted color and LBP histogram models based on particle filter to achieve better performance for human tracking. Particle filter is used to estimate the posterior density function and human state using a set of *N* weighted particles $\{X_t{}^k, w_t{}^k\}_{k=1:N}$, where $X_t{}^k$ and $w_t{}^k$ are the state and the weight of *k*th particle at time *t* respectively. In Ref. 80, they also used particle filter to track moving objects in outdoor environment based on color feature.

3.3.3. *Mean shift algorithm*

Mean shift algorithm is used to find the peak of the confidence map near the object's old position.[46] Diwakar[57] used mean shift algorithm to track the objects. From the fusion of color and texture object appearance models, a new location of the object is estimated using Bhattacharyya coefficient. Mean shift algorithm can be used to solve occlusion problem[22] and help to track broken object due to color similarity between foreground object and background image.[46] The main disadvantages of using mean shift algorithm are that, the window size could not adapt to the changes in the object size across the frames as shown in Fig. 5 and as well as to the changes in orientation as shown in Fig. 6.[82]

3.3.4. *CamShift algorithm*

CamShift algorithm solves the problem in mean shift algorithm by adapting the window size with size and rotation of the target. The CamShift works by finding which

Fig. 5. Mean shift algorithm unable to adapt with changes in scale.[81]

Fig. 6. Mean shift algorithm unable to adapt with changes in orientation.[82]

Fig. 7. Person tracking using CamShift algorithm from a moving camera in outdoors environment.[82]

probability density increasing maximum and detect the real position of the moving object. The similarity function is used to describe the similarity of the target model and candidate model. Both of them are described as the probability density distribution. In Ref. 45, they used edge distribution based CamShift algorithm in their object tracking system. However, the method is not suitable for a low resolution video because some of the edge might be blurred, thus lead to a false tracking. Sachdeva and Birok[82] further emphasized that the algorithm is computationally fast. However, it fails to track the object when the color of foreground object is similar to the background image as shown in Fig. 7.

3.3.5. *Euclidean distance*

In Refs. 27, 28 and 46, they limit the number of possible candidates for object matching by considering the candidates that are close to the target blob in the current frame only. They used Euclidean distance to compute the distance between centroid of candidate blobs and target blob. Euclidean distance computes the distance between two points in Euclidean space. In Refs. 14 and 30, they applied Euclidean distance to compute the distance between histograms.

3.3.6. *Bhattacharyya coefficient*

Bhattacharyya coefficient measure the amount of overlap between two statistical samples or the relative closeness of the two samples. The similarity function defines the distance between target appearance model and candidate appearance model. In Refs. 17, 47 and 57, they utilized Bhattacharyya coefficient to match the target blobs with the candidate blobs based on bounding box intersection i.e., if the bounding box target blob intersects with the candidate blob, Bhattacharyya coefficient is computed based on Eq. (5). The target is assigned to the candidate blob with the highest coefficient and the histogram is updated. However, if the target blob is in a merged condition or in occlusion, multiple candidate blobs are matched with one target blob. The candidate blobs that matched to the target blob are put into a merge state, but the model histograms are not updated.

$$\hat{\rho}(y) \equiv \rho[\hat{\boldsymbol{p}}(\boldsymbol{y}), \hat{\boldsymbol{q}}] = \sum_{u=1}^{m} \sqrt{\hat{p}_u(\boldsymbol{y}), \hat{q}_u} \tag{5}$$

The terms $\hat{\boldsymbol{q}} = \{\hat{q}_u\}_{u=1\ldots m}$, and $\hat{\boldsymbol{p}}(\boldsymbol{y}) = \{\hat{p}_u(\boldsymbol{y})\}_{u=1\ldots m}$ are the probabilities estimated from the *m*-bins histograms of target and candidate blobs. The similarity measure using Bhattacharyya coefficient is not discriminative enough because the object appearance is reduced to a global histogram.[83] In Ref. 17, they modified Bhattacharyya coefficient in Eq. (5) to compute the distance between two color histograms *k* and *q* based on Eq. (6).

$$D(k,q) = \sqrt{1 - \sum_{i=1}^{m} \sqrt{\hat{k}_i \hat{q}_i}} \tag{6}$$

3.3.7. *Bhattacharyya distance*

Bhattacharyya distance is closely related to Bhattacharyya coefficient. It computes the similarity of two discrete or continuous probability distributions. Sugandi[80] used Bhattacharyya distance based on (2.16) to compute the similarity between color histogram of target and candidate blobs, where H_1 and H_2 are histograms and *i* is index of bins. Rahimi *et al.*[25] utilize Bhattacharyya distance between human model and the human observations weighted color and LBP histograms as a criterion for occlusion detection.

$$d(H_1, H_2) = \sum_{i} \sqrt{H_1(i) \cdot H_2(i)} \tag{7}$$

3.3.8. *Homography transformation*

Homography transformation can only be used for overlapping views. It links the position of the same points between a pair of cameras. This method needs at least four points between two overlapped views to estimate the homography matrix. In Ref. 25, they solved the occlusion problem under overlapping cameras view based on homography transformation. The occluded humans are detected at each camera individually and the homography transformation is applied to locate the occluded persons at occluded view.

3.3.9. *CSK tracker*

Methods that apply tracking by detection involve sampling strategy to train a set of discriminative classifier. The processing requires more computational effort for feature extraction, training and detection when the sampling is done individually. A set of samples contains redundant information if they are sampled with an overlap.[49] Circulant Structure of Tracking-by-Detection with Kernels (CSK) Tracker exploits the redundancy of samples to provide much faster computation. It relies on a kernelized least square classifier. The feature is extracted using Fast Fourier Transform (FFT), thereby low computational cost.[84] However, it is not reliable for scale variation, fast target motions and long-term occlusions.

Table 2. Tracking algorithms with their advantages and limitations.

Algorithms	Advantages	Limitations	Ref.
Kalman Filter	Prediction may reduce computational cost because if the predictor tells the position of an object	Not reliable for non-linear or complex motion	[24, 75]
Particle Filter	• Very suitable for tracking multiple object • Give good estimations for state variables that are non-linear	Computational is complex and expensive since all the particles needed to achieve a required approximation accuracy	[77–79]
Mean Shift	Can locate object after occlusion	• Window size for tracking is not changing • Unable to adapt to changes in orientation	[22, 82]
CamShift	• Solve the problem in mean shift algorithm by adapting the window size with size and rotation of the target • Computationally fast	Not reliable when the color of foreground object is similar to the background image	[45, 82]
Euclidean distance	Can track the object across frames	Not reliable when a new object enter camera's field of view	[15, 30, 46]
Bhattacharyya coefficient	Defines the distance between target appearance model and candidate appearance model	• Similarity measure is not discriminative enough because the appearance of the target object is reduced to a global histogram • Not reliable for correlating gray-scale objects	[17, 47, 57, 67, 86]
Bhattacharyya distance	Computes the similarity of two discrete or continuous probability distributions	Not discriminative enough because the appearance of the target object is reduced to a global histogram	[46, 67]
Homography transformation	Solve occlusion problem in multiple cameras	Not reliable for network cameras with non-overlapping views	[25]
CSK Tracker	Computational fast	Not reliable for scale variation, fast target motions and long-term occlusions	[84]
Support Vector Machines (SVMs)	• Gives accurate localization and flexible for target modelling	• Takes a lot of time for training • Under-training or over-training effect accuracy • Need correct number of samples	[19, 85]

Table 3. Analysis in multiple object tracking for single camera.

Types	Features	Tracking Algorithm	H	V	O	I	C	OC	A	R	Ref.	Year
Single feature	Color	Bhattacharyya distance	√	X	X	√	X	√	X	√	[17]	2010
			√	X	√	√	√	√	X	√	[53]	2012
		Bhattacharyya distance, Particle Filter	√	X	√	X	X	√	√	X	[80]	2010
		Mean-shift	X	√	√	X	√	X	√	X	[59]	2009
		Color Sparse Generative Model (SGM) tracker	√	X	√	X	√	√	X	√	[87]	2014
	Intensity	Mean Intensity Similarity Measurement	√	X	√	X	X	X	X	√	[13]	2009
	Edge	Mean-shift	√	√	√	X	X	√	X	√	[61]	2011
		CamShift	√	X	X	√	X	X	X	√	[45]	2010
	Texture	Naive Bayesian Classifier	√	X	X	√	X	X	X	√	[77]	2014
		Particle Filter	X	√	√	X	X	X	X	√	[89]	2013
	Position	Kalman Filter	√	X	√	X	X	√	X	√	[24]	2008
			√	√	√	X	√	√	X	√	[60]	n.d
Multiple features	Color, Gait period, Position	Kalman Filter	√	√	√	√	√	√	X	X	[23]	2011
	Color, Texture	Mean-shift	X	√	√	X	X	X	√	X	[57]	2013
			√	X	√	√	X	X	√	X	[90]	2013
	Color, Edge	Bhattacharyya coefficient, Mean-shift	√	√	√	X	X	√	X	√	[65]	2007
	Color, Position	Bhattacharyya coefficient, Euclidean distance, Mean-shift	√	X	√	X	X	X	X	√	[46]	2013
		Absolute distance	√	√	√	√	X	√	X	√	[15]	2013
		Kalman filter	X	√	√	X	X	X	X	√	[72]	2014
		Normalized Geometric Distance, Euclidean distance	√	X	√	X	X	√	X	√	[27] [28]	2014
	Position, Shape, Color	Adaptive Filter, Hierarchy matching	√	√	√	X	√	√	X	√	[48]	2006
	Color, Texture, Oriented gradient	Bhattacharyya coefficient, Particle Filter	√	X	√	X	√	√	√	X	[25]	2013
	Edge, Texture, Color Intensity	Particle Filter	√	X	√	√	X	√	√	X	[91]	2013

Note: Table notes.
H: Paper include human tracking, V: Paper include vehicle Tracking, O: Dataset include outdoor environment, I: Dataset include indoor environment, C: Dataset include crowded cases (more than 3 objects at the same time), OC: Dataset include Occluded Objects, A: Paper solve accuracy problem, R: Paper solve speed (real-time) problem.

Table 4. Analysis in multiple object tracking for multiple cameras.

Types	Features	Tracking Algorithm	H	V	O	I	C	OC	OV	NOV	A	R	Ref.	Year
Single feature	Color	Novel similarity measurement	√	X	√	X	X	√	X	√	X	√	[24]	2008
		Support Vector Machines (SVMs)	√	X	√	√	√	√	X	√	X	√	[53]	2012
		Extended CSK Tracker	√	X	√	X	X	√	X	√	X	√	[51]	2014
Multiple features	Color, Direction	Stochastic Matching	√	X	√	X	√	√	X	√	X	√	[54]	2011
	Intensity, Shape	Mean Intensity Similarity Measurement	√	X	√	√	X	X	X	√	X	√	[13]	2009
	Color, Height, Speed, Travel time, Moving direction	Overall Similarity Score	√	X	X	√	X	√	X	√	X	√	[66]	2010
	Shape, Color, Texture, Height	Median Similarity Measurement	√	X	X	√	X	X	√	√	X	√	[26]	2011
	Travel time, Color, Texture	Objective Function, Camera Link Model	√	X	√	X	X	X	X	√	X	X	[55]	2012
	Color Gait period, Centroid position, Moving direction	Bayes Rule, Homography, Similarity Distance Measurement	√	√	√	√	√	√	√	√	X	√	[22]	2011
	Speed, Direction of motion	Graph Model	√	X	√	X	X	X	X	√	X	√	[56]	2013
	Color, Time, Position	Bhattacharyya distance	√	√	√	X	X	X	X	√	X	√	[46]	2013
	Position, Color, Edge	Overall Similarity Score	√	√	√	X	X	√	X	√	X	√	[27]	2014
	Position, Color, Edge, Texture	Overall Similarity Score	√	√	√	X	X	√	X	√	√	√	[28]	2014

Note: Table notes.

H: Paper include human tracking, V: Paper include vehicle Tracking, O: Dataset include outdoor environment, I: Dataset include indoor environment, C: Dataset include crowded cases (more than 3 objects at the same time), OC: Dataset include Occluded Objects, OV: Dataset include Overlapping Camera Field of Views, NOV: Dataset include Non-overlapping Camera Field of Views, A: Paper solve accuracy problem, R: Paper solve speed (real-time) problem.

3.3.10. *Support Vector Machines* (*SVMs*)

SVMs are a group of supervised learning models that can be applied to classification or regression. SVMs has been used in many tracking algorithms because it provides an accurate localization and flexible for target modelling.[19] However, this algorithm will give a poor performance if the number of features is greater than the number of samples. SVMs take a lot of time for training the images for several of views and the algorithm is quite complex. Under-training or over-training the images might affect the accuracy of the tracking. It only gives efficient result for small training samples.[85] Table 2 shows a list of tracking algorithms that are usually used for single camera view and multiple cameras.

Tables 3 and 4 listed the analysis done on 21 papers and 12 papers that were published on multiple objects tracking within single camera view and multiple cameras views respectively.

4. Conclusion

This paper reviews on the available techniques used for object detection and visual tracking for single camera and multiple cameras in video surveillance applications. From the published literature, it can be concluded that the object detection consists of four consecutive processes that are: (1) Pre-processing, (2) Object segmentation, (3) Mathematical morphology and (4) Blob extraction. In the case of visual tracking, the tracking process can be divided into two main steps: (1) Modelling and (2) Tracking. In the modelling stage, different type of features has been used to build the object appearance model. The advantages and limitations of available features are tabulated into Table 1. Detailed summaries on related algorithms used for tracking multiple moving objects within single view and multiple cameras views have also been discussed. Each tracking algorithms advantages and limitations have been tabulated into Table 2. A comprehensive analysis from 21 papers that were published on multiple object tracking within single camera view and from 12 papers for multiple camera views have also been included in this paper.

Based on the literature survey, most of the available techniques proposed by the earlier researchers can perform object detection and tracking either within single camera view or across multiple cameras. However, most of them failed to encounter trade-off problem between accuracy and speed. Although the accuracy of the trackers is very good, they often impractical because of their high computational requirements and vice versa. Thus, to achieve an optimal trade-off, adaptive object detection and tracking method is essential to achieve a real-time and reliable surveillance system. It is due to this reason that the main aim of this paper is to provide a valuable insight into the related areas of the related research topic in video surveillance and to promote new research.

This research uses more features for modeling the object's appearance including multiple-color histograms consist of Hue color and YCbCr color because single histogram may not capture local color characteristics of the object efficiently. Besides, edge feature is used as an additional feature to texture and shape features for modeling

Table 5. Summary of selected methods and improvements of existing methods.

Level	Process	Selected Method	Improvements	Comments
Object Detection	Background Modeling	Gaussian Mixture Model	—	Fast computation with appropriate number of Gaussian
	Object Segmentation	Background Subtraction	—	1. Has good performance for extracting image pixels of foreground objects 2. Computational fast because it relies on the pixel values of the image
	Morphological Operation	Opening and Closing Operators	—	Smoothened the noise in the background from binary image while keeping the size of foreground objects
	Blob Extraction	Connected Components Labeling Algorithm	1. Connecting broken mask for large object based on nearest Euclidean distance 2. Quick shadow removal based on human head width	Fast computational time for real-time performance based on simple algorithm implementation with an accurate blob extraction
Object Tracking	Algorithm for Modeling	Local Binary Pattern (LBP)	Use 5 bins to cluster the LBP descriptors	1. Robust to illumination changes due to its gray-scale invariance characteristic and deformation 2. Improve accuracy during correspondence management
		Color Histogram	Use multiple-color histograms. The features are normalized by the size of the segmented region to adapt with scale variation. 1. Mean value of Hue color 2. Mean value from two minimum values of (Y, Cb, Cr) channels	1. Represent distribution of color in an image using multi-color space because single histogram may not capture local characteristics of the object efficiently 2. Insensitive to scale variation
Object Tracking	Algorithm for Modeling	Canny Edge Detection	Edge feature computed from white pixels in binary image normalized by segmented region's size. Then, fuse edge feature with texture feature.	1. Robust to illumination variation 2. It gives more accurate result during correspondence management 3. Insensitive to scale variation
		Patch extraction	Extraction of valid regions from human body (upper part and lower part) based on human head width.	Reliable for variation in pose and viewing angle.
	Algorithms for Tracking	Euclidean Distance	Compute the minimum distance of object centroid in the previous and current frames for single-camera tracking.	1. Fast computation and simple implementation 2. No interchange of label when the objects are entering and exiting almost at the same time
		Normalized Geometric Distance	Compute distance between object appearance features of objects for correspondence management.	Improve tracking accuracy during similarity measurement between existing objects and new object.

the object's appearance. Combination of Hue color, edge, texture and shape features are used to increase the accuracy of tracking for multiple camera views caused by incorrect object recognition due to variation of illumination condition. They are used in modeling the object's appearance since they are insensitive to illumination variation.

For multi-cameras environment, the cameras are being located at different locations and have different lighting conditions. Thus tracking moving objects across multiple cameras view is more challenging than a single camera view. For non-overlapping view, there is no spatial-continuity between the cameras which is more challenging than overlapping view. Spatial-temporal features such as motion direction, speed and travel time are not reliable for tracking moving objects in non-overlapping view when they are travelling into the same camera view. This is due to inconsistent objects' motion and unknown travel time when the objects enter the same camera view immediately or at a moment later after they have left the scene. Thus, in this research, object appearance-based features are used to find the correspondence between the object that is entering and the candidate objects from the existing database which is more reliable for this case.

For single-camera object tracking, only position feature is used to reduce the computational cost. Only certain parts of the image must be processed, thus, reducing the amount of data contained in the video led to fast computation. The proposed method improves the existing techniques that rely on the objects' appearance features to track the objects across the frames which are computationally expensive. The list of selected approaches and some improvements made in this research are summarized in Table 5.

Acknowledgments

This work is supported by the Fundamental Research Grant Scheme (FRGS) Research Project FRGS13–030-0271 under the Ministry of Higher Education of Malaysia (MOHE). This work is also collaboration between IIUM and Royal Malaysia Police (RMP).

References

1. A. Koutsia *et al.*, Traffic monitoring using multiple cameras, homographies and multi-hypothesis tracking, in *3DTV Conf.* (Kos Island: IEEE, 2007), pp. 1-4.
2. G. Geiger, T. Hazel and D. Vogt, Integrated SCADA-based approach for pipeline security and operation, in *Record of Conference Papers Industry Applications Society 57th Annual Petroleum and Chemical Industry Conference* (*PCIC*) (San Antonio, TX: IEEE, 2010), pp. 1–8.
3. Y. M. Mustafah, A. W. Azman, A. Bigdeli and B. C. Lovell, An automated face recognition system for intelligence surveillance: smart camera recognizing faces in the crowd, in *First ACM/IEEE Int. Conf. on Distributed Smart Cameras* (*ICDSC '07*) (Vienna: IEEE, 2007), pp. 147–152.
4. C. Mauser, W. Granzer and W. Kastner, Integrating CCTV systems into BACnet, in *IEEE 16th Conf. on Emerging Technologies & Factory Automation* (*ETFA*) (Toulouse: IEEE, 2011), pp. 1–4.

5. I. Darker, A. Gale, L. Ward and A. Blechko, Can CCTV reliably detect gun crime?, in *41st Annual IEEE Int. Carnahan Conf. on Security Technology* (Ottawa, Ont.: IEEE, 2007), pp. 264–271.
6. S. Zambanini, P. Blauensteiner and M. Kampel, Automated multi-camera surveillance for the prevention and investigation of bank robberies in Austria: A case study, in *3rd Int. Conf. on Crime Detection and Prevention* (*ICDP 2009*) (London: IET, 2009), pp. 1–6.
7. Y. M. Mustafah, A. Bigdeli, A. W. Azman and B. C. Lovell, Smart cameras enabling automated face recognition in the crowd for intelligent surveillance system, in *Proc. of the Security Technology Conference. Recent Advances in Security Technology* (*RNSA'07*) (2007), pp. 310–318.
8. A. M. Ibrahim, *Automated Tracking System for Video Surveillance System*. Master thesis, International Islamic University Malaysia (2013).
9. R. A. Lotufo, A. D. Morgan and A. S. Johnson, Automatic number-plate recognition, in *IEE Colloquium on Image Analysis for Transport Applications* (London: IET, 1990), pp. 6/1–6/6.
10. A. Williem, V. Madasu, W. Boles and P. Yarlagadda, A context-based approach for detecting suspicious behaviours, in *2009 Digital Image Computing: Techniques and Applications* (*DICTA '09*) (Melbourne, VIC: IEEE, 2009), pp. 146–153.
11. Y. M. Mustafah and A. W. Azman, Out-of-plane rotated object detection using patch feature based classifier, *Journal of Procedia Engineering, Elsevier. Int. Symp. on Robotics and Intelligent Sensors* (*IRIS*) **41** (2012) 170–174.
12. M. Sonka, V. Hlavac and R. Boyle, *Image Processing, Analysis, and Machine Vision* (United States of America: Thomson, 2008).
13. M. O. Mehmood and A. Khawaja, Multi-camera based human tracking with non-overlapping fields of view, in *Fifth Int. Conf. on Image and Graphics* (2009), p. 313.
14. V. Takala and M. Pietikainen, Multi-object tracking using color, texture and motion, in *IEEE Conf. on Computer Vision and Pattern Recognition* (*CVPR '07*) (Minneapolis, MN: IEEE, 2007), pp. 1–7.
15. D. Jansari, S. Parmar and G. Saha, Real-time object tracking using color-based probability matching, in *Int. Conf. on Signal Processing, Computing and Control* (*ISPCC*) (Solan: IEEE, 2013), pp. 1–6.
16. M. Casares and S. Velipasalar, Light-weight salient foreground detection for embedded smart cameras, in *Second ACM/IEEE Int. Conf. on Distributed Smart Cameras* (*ICDSC*) (Standford, CA: IEEE, 2008), pp. 1–7.
17. O. Javed, Z. Rasheed, O. Alatas and M. Shah, KNIGHTM: A real time surveillance system for multiple overlapping and non-overlapping cameras, in *Int. Conf. on Multimedia and Expo* (*ICME '03*), Vol. 1 (2003), pp. I-649-52.
18. O. Javed, K. Shafique and M. Shah, A hierarchical approach to robust background subtraction using color and gradient information, *Workshop on Motion and Video Computing* (2009), pp. 22–27.
19. P. Zhao, R. Zhang and T. Shibata, Real-time object tracking algorithm employing on-line support vector machine and multiple candidate regeneration, *Artificial Intelligence and Soft Computing*, *Lecture Notes in Computer Science*, Vol. **7267** (Springer Berlin Heidelberg, 2012), pp. 617–625.
20. M. Heikkilä and M. Pietikäinen, A texture-based method for modelling the background and detecting moving objects, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28** (2006) 657–662.
21. R. Zhang and J. Ding, Object tracking and detecting based on adaptive background subtraction, *Procedia Engineering* **29** (2012) 1351–1355.
22. D. T. Lin and K. Y. Huang, Collaborative pedestrian tracking and data fusion with multiple cameras, *IEEE Transactions on Information Forensics and Security* **6**(4) (2011) 1432–1444.

23. C. Stauffer and W. Grimson, Adaptive background mixture models for real-time tracking, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Vol. 2 (1999), pp. 246–252.
24. Y. Cai, K. Huang and T. Tan, Matching tracking sequences across widely seperated cameras, in *ICIP 2008* (IEEE, 2008), pp. 765–768.
25. S. Rahimi, A. Aghagolzadeh and H. Seyedarabi, Three camera-based human tracking using weighted color and cellular LBP histograms in a particle filter framework, in *21st Iranian Conf. on Electrical Engineering* (*ICEE*) (Mashhad: IEEE, 2013), pp. 1–6.
26. J. Sherrah, D. Kamenetsky and T. Scoleri, Evaluation of similarity measures for appearance-based multi-camera matching, in *2011 Fifth ACM/IEEE Int. Conf. on Distributed Smart Cameras* (*ICDSC*) (Ghent: IEEE, 2011), pp. 1–6.
27. N. N. A. Aziz, Y. M. Mustafah, A. A. Shafie, M. A. Rashidan and N. A. Zainuddin, Real-time tracking using edge and color feature, in *Int. Conf. on Computer and Communication Engineering* (*ICCCE*) (Kuala Lumpur: IEEE, 2014), pp. 247–250.
28. N. N. A. Aziz, Y. M. Mustafah, A. W. Azman, N. A. Zainuddin and M. A. Rashidan, Features selection for multi-camera tracking, in *Int. Conf. on Computer and Communication Engineering* (*ICCCE*) (Kuala Lumpur: IEEE, 2014), pp. 243–246.
29. F. Hafiz, A. A. Shafie, O. O. Khalifa and M. H. Ali, Foreground segmentation-based human detection with shadow removal, in *Proc. of 2010 Int. Conf. on Computer and Communication Engineering* (*ICCCE*) (Kuala Lumpur: IEEE, 2010), pp. 1–6.
30. M. Huang, G. Chen, G.-F. Yang and R. Cao, An algorithm of the target detection and tracking of the video, *Procedia Engineering* **29** (2012) 2567–2571.
31. N. A. Zainuddin, Y. M. Mustafah, A. W. Azman, M. A. Rashidan and N. N. A. Aziz, Analysis on background subtraction for street surveillance, in *Int. Conf. on Computer and Communication Engineering* (*ICCCE*) (Kuala Lumpur: IEEE, 2014), pp. 236–239.
32. A. A. Shafie, M. A. Ibrahim and M. M. Rashid, Smart objects identification system for robotic surveillance, in *International Journal of Automation and Computing* **11**(1) (2014) 59–71.
33. F. H. B. K. Zaman, *Automated Human Recognition and Tracking for Video Surveillance System*, Master dissertation, International Islamic University Malaysia (2010).
34. C. X. Wang and W. J. Zhang, A robust algorithm for shadow removal of foreground detection in video surveillance, in *Proc. of 2009 Asia-Pacific Conf. on Information Processing* (*APCIP 2009*) (Washington, DC, USA), Vol. 2 (IEEE, 2009), pp. 422–425.
35. M. A. Ibrahim, A. A. Shafie and M. M. Rashid, Human identification system based on moment invariant features, in *Int. Conf. on Computer and Communication Engineering* (*ICCCE*) (Kuala Lumpur: IEEE, 2012), pp. 216–221.
36. Y. M. Mustafah and A. W. Azman, Skin region detector for real time face detection system, in *Int. Conf. on Computer and Communication Engineering* (*ICCCE*) (Kuala Lumpur: IEEE, 2012), pp. 653–658.
37. Y. M. Mustafah, A. Bigdeli, A. W. Azman and B. C. Lovell, Face detection system design for real time high resolution smart camera, in *Third ACM/IEEE Int. Conf. on Distributed Smart Cameras* (*ICDSC*) (Como, Italy: IEEE, 2009), pp. 1–6.
38. Y. M. Mustafah, T. Shan, A. W. Azman, A. Bigdeli and B. C. Lovell, Real-time face detection and tracking for high resolution smart camera system, in *9th Biennial Conf. of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications* (Glenelg, Australia: IEEE, 2007), pp. 387–393.
39. A. A. Shafie, F. Hafiz and M. H. Ali, Motion detection techniques using optical flow, *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering* **3**(8) (2009) 1561–1563.
40. T. Brox, A. Bruhn, N. Papenberg and J. Weickert, High accuracy optical flow estimation based on a theory for warping, in *Proc. 8th European Conf. on Computer Vision*, eds. T. Pajdla and J. Matas, *Springer LNCS*, Vol. **3024** (Prague, Czech Republic, 2004), pp. 25–36.

41. N. Ibrahim, S. S. Mokri, L. Y. Siong, M. M. Mustafa and A. Hussain, Snatch theft detection using low level features, in *Proc. of the World Congress on Engineering 2010* (*WCE 2010*) (London, UK: IEEE, 2010), pp. 1–5.
42. M. S. Kemouche and N. Auof, A Gaussian mixture based optical flow modelling for object detection, in *3rd Int. Conf. on Crime Detection and Prevention* (*ICDP 2009*) (London: IEEE, 2009), pp. 1–6.
43. F. Ranchin and F. Dibos, Moving objects segmentation using optical flow estimation (2003).
44. B. Gagan and S. Mullur, Vehicle detection & tracking, n.d.
45. Y. Yang, Z. Wang, D. Sun, M. Zhang and N. Cheng, Automatic object tracking using edge orientation histogram based CamShift, in *Third Int. Conf. on Information and Computing* (*ICIC 2010*) (Wuxi, Jiang Su: IEEE, 2010), pp. 231–234.
46. H. H. Hsu, W. M. Yang and T. K. Shih, People tracking in a multi-camera environment, in *Conference Anthology* (China: IEEE, 2013), pp. 1–4.
47. Y. Wang, S. Velipasalar and M. C. Gursoy, Wide-area multi-object tracking with non-overlapping camera views, in *Int. Conf. on Multimedia and Expo* (*ICME*) (Barcelona: IEEE, 2011), pp. 1–6.
48. Q. Zhou and J. K. Aggarwal, Object tracking in an outdoor environment using fusion of features and cameras, *Image and Vision Computing* **24**(11) (2006) 1244–1255.
49. M. Danelljan, *Visual Tracking* (Linköping University Electronic Press, 2013).
50. F. S. Khan, J. van de, Weijer and M. Vanrell, Modulating shape features by color attention for object recognition, *IJCV* **98**(1) (2012) 49–64.
51. M. Danelljan, F. S. Khan, M. Felsberg and J. v. Weijer, Adaptive color attributes for real-time visual tracking, in *IEEE Conf. on Computer Vision and Pattern Recognition* (Columbus, OH: IEEE, 2014), pp. 1090–1097.
52. J. Van De Weijer, C. Schmid, J. Verbeek and D. Larlus, Learning color names for real-world applications, *IEEE Transactions on Image Processing* **18**(7) (2009) 1512–1523.
53. Q. Huang, J. Yang and Y. Qiao, Person re-identification across multi-camera system based on local descriptors, in *2012 Sixth Int. Conf. on Distributed Smart Cameras* (*ICDSC*) (Hong Kong: IEEE, 2012), pp. 1–6.
54. X. Chen, K. Huang and T. Tan, Direction-based stochastic matching for pedestrian recognition in non-overlapping cameras, in *18th IEEE Int. Conf. on Image Processing* (IEEE, 2011), pp. 2065–2068.
55. C. T. Chu, J. N. Hwang, J. Y. Yu and K. Z. Lee, Tracking across nonoverlapping cameras based on the unsupervised learning of camera link models, in *2012 Sixth Int. Conf. on Distributed Smart Cameras* (*ICDSC*) (Hong Kong, 2012), pp. 1–6.
56. J. Xu, V. Jagadeesh, Z. Ni, S. Sunderrajan and B. S. Manjunath, Graph-based Topic-focused retrieval in distributed camera network, in *IEEE Transactions on Multimedia* (2013), pp. 1–13.
57. M. Diwakar, P. K. Patel, K. Gupta and C. Chauhan, Object tracking using joint enhanced color-texture histogram, in *Proc. of the 2013 IEEE Second Int. Conf. on Image Information Processing* (*ICIIP-2013*) (Shimla: IEEE, 2013), pp. 160–165.
58. C. Li, X. Wang and X. Xu, Robust object tracking with adaptive fusion of color and edge strength local mean features based on particle filter, in *2009 Int. Forum on Information Technology and Applications* (*IFITA '09*), Vol. 3 (2009), pp. 285–289.
59. B. Li, Z.-Y. Zeng and Z.-R. Wu, Multi-object tracking based on improved mean-shift algorithm, in *2nd Int. Congress on Image and Signal Processing* (*CISP '09*) (Tianjin: IEEE, 2009), pp. 1–5.
60. G. Bansal and S. Mullur, Vehicle detection & tracking, n.d, pp. 1–5.
61. M. Murshed, M. H. Kabir and O. Chae, Moving object tracking — An edge segment based approach, *International Journal of Innovative Computing Information and Control* **7**(7(A)) (2011) 3963–3979.

62. S. Birchfield, Elliptical head tracking using intensity gradients and color histograms, in *Proc. CVPR* (1998), pp. 232–237.
63. D. Comaniciu, V. Ramesh and P. Meer, Kernel-based object tracking, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **25**(5) (2003) 564–577.
64. S. T. Birchfield and S. Rangarajan, Spatial histograms for region-based tracking, *ETRI Journal* **29**(5) (2007) 697–699.
65. W. Liu and J. Y. Zhang, Real time object tracking using fused color and edge cues, in *9th Int. Symp. on Signal Processing and Its Applications* (*ISSPA 2007*) (Sharjah: IEEE, 2007), pp. 1–4.
66. Y. Wang, L. He and S. Velipasalar, Real-time distributed tracking with non-overlapping cameras, in *IEEE 17th Int. Conf. on Image Processing* (IEEE, 2010), pp. 697–700.
67. H. Wang, D. Suter and K. Schindler, Effective appearance model and similarity measure for particle filtering and visual tracking, in *Computer Vision* (*ECCV 2006*) (Austria: Springer Berlin Heidelberg, 2006), pp. 606–618.
68. R. C. Gonzalez, R. E. Woods and S. L. Eddins, *Digital Image Processing Using Matlab* (New Delhi: McGraw Hill, 2010).
69. C. Shan, S. Gong and P. W. McOwan, Robust facial expression recognition using local binary patterns, in *IEEE Int. Conf. on Image Processing* (*ICIP 2005*) (IEEE, 2005), pp. II-370-3.
70. N. Dalal and B. Triggs, Histogram of oriented gradients for human detection, in *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* (*CVPR 2005*), Vol. 1 (San Diego, CA, USA: IEEE, 2005), pp. 886–893.
71. P. Li, A clustering based color model and fast algorithm for object tracking, in *Proc. of the 18th Int. Conf. on Pattern Recognition* (*ICPR'06*) (Hong Kong: IEEE, 2006), pp. 671–674.
72. D. Gambhir and M. Manchanda, Adaptive threshold based segmentation for video object tracking, in *IEEE Int. Advance Computing Conf.* (*IACC*) (Gurgaon: IEEE, 2014), pp. 1127–1132.
73. P. Vora and B. Oza, A survey on K-mean clustering and particle swarm optimization, *International Journal of Science and Modern Engineering* **1**(3) (2013) 24–26.
74. S. Bilal, R. Akmeliawati, M. J. E. Salami, A. A. Shafie and E. M. Bouhabba, A hybrid method using haar-like and skin-color algorithm for hand posture detection, recognition and tracking, in *Int. Conf. on Mechatronics and Automation* (*ICMA*) (Xi'an: IEEE, 2010), pp. 934–939.
75. L. Bazzani, D. Bloisi and V. Murino, A comparison of multi hypothesis Kalman filter and particle filter for multi-target tracking, *Performance Evaluation of Tracking and Surveillance Workshop at CVPR* (2009), pp. 47–54.
76. E. Orhan, Particle filtering (2012).
77. A. Pathak and E. Singh, Comparative study on filtering techniques of digital image processing, *Advance in Electronic and Electric Engineering* **4**(6) (2014) 669–674.
78. L. Hong and K. Xue, A spatial domain multiresolutional particle filter, *Mediterranean Conf. on Control & Automation* (*MED '07*) (Athens: IEEE, 2007), pp. 1–6.
79. H. Zhu, H. Zhao, D. Liu and C. Song, A new improved filter for target tracking: Compressed iterative particle filter, *Natural Science* **3**(4) (2011) 301–306.
80. B. Sugandi, H. Kim, J. K. Tan and S. Ishikawa, Tracking of multiple moving objects under outdoor environment using color-based particle filter, in *3rd Int. Conf. on Computer Science and Information Technology* (*ICCSIT*) (Chengdu: IEEE, 2010), pp. 103–107.
81. M. Alexander and K. Abid, Meanshift and Camshift. [Online]. http://opencv-python-tutroals.readthedocs.org/en/latest/py_tutorials/py_video/py_meanshift/py_meanshift.html (2013).
82. C. Sachdeva and R. Birok, Real time object tracking using different mean shift techniques — A review, *International Journal of Soft Computing and Enngineering* (2013) 98–102.

83. C. Yang, R. Duraiswami and L. Davis, Multiple object tracking via a hierarchical particle filter, in *Int. Conf. on Computer Vision* (2005), pp. 212–219.
84. J. F. Henriques, R. Caseiro, P. Martins and J. Batista, Exploiting the circulant structure of tracking-by-detection with kernels, *Computer Vision* (*ECCV 2012*) (Springer Berlin Heidelberg, 2012), pp. 702–715.
85. P. M. Shah, Face detection from images using support vector machine, in *Master's Project.* Paper 321 (2012).
86. M. S. Khalid, M. U. Ilyas, M. S. Sarfaraz and M. A. Ajaz, Bhattacharyya coefficient in correlation of gray-scale objects, *Journal of Multimedia* **1**(1) (2006) 56–61.
87. H. Zheng, C. Tian and W. Wei, Visual tracking based on the color attention preserved sparse generative object model, in *IEEE Int. Conf. on Orange Technologies* (*ICOT*) (Xi'an, China: IEEE, 2014), pp. 1–4.
88. C. Lin and C.-M. Pun, Object tracking using dimension reduction of descriptive features, in *11th Int. Conf. on Computer Graphics, Imaging and Visualization* (*CGIV*) (Singapore: IEEE, 2014), pp. 73–77.
89. N. Yao, Z.-H. Liu and F. Qian, A new target tracking method based on image patches, in *Proc. of the 2013 Int. Conf. on Machine Learning and Cybernetics* (Tianjin: IEEE, 2013), pp. 865–870.
90. D. Y. Lee, J. Y. Sim and C. S. Kim, Fast object tracking using color histograms and patch differences, in *20th IEEE Int. Conf. on Image Processing* (*ICIP*) (Melbourne, VIC: IEEE, 2013), pp. 3905–3908.
91. W. Chen, L. Cao, J. Zhang and K. Huang, An adaptive combination of multiple features for robust tracking in real scene, in *2013 IEEE Int. Conf. Computer Vision Workshops* (Sydney, NSW: IEEE, 2013), pp. 129–136.