The State-of-the-Art in Visual Object Tracking

Anand Singh Jalal
Department of Computer Engineering & Applications
GLA University, Mathura, India - 281406
E-mail: anandsinghjalal@gmail.com

Vrijendra Singh
Indian Institute of Information Technology, Allahabad, India - 211012
E-mail: vrij@iiita.ac.in

Overview paper

Keywords: object tracking, object modelling, moving object detection

Received: May 27, 2011

There is a broad range of applications of visual object tracking that motivate the interests of researchers worldwide. These include video surveillance to know the suspicious activity, sport video analysis to extract highlights, traffic monitoring to analyse traffic flow and human computer interface to assist visually challenged people. In general, the processing framework of object tracking in dynamic scenes includes the following stages: segmentation and modelling of interesting moving object, predicting possible location of candidate object in each frame, localization of object in each frame, generally through a similarity measure in feature space. However, tracking an object in a complex environment is a challenging task. This survey discusses some of the core concepts used in object tracking and present a comprehensive survey of efforts in the past to address this problem. We have also explored wavelet domain and found that it has great potential in object tracking as it provides a rich and robust representation of an object.

Povzetek: Podan je pregled metod vizualnega sledenja objektov .

1 Introduction

As the technology is advancing with rapid pace, the cost of video cameras and digital media storage is affordable. In recent years, we have seen a remarkable increase in the amount of video data recorded and stored around the world. In order to process all these video data, there is a growing demand of automatically analyze and understand the video contents. One of the most fundamental processes in understanding video contents is visual object tracking, which is the process of finding the location and dynamic configuration of one or more moving objects in each frame (image) of a video [1].

There is a broad range of applications of object tracking that motivate the interests of researchers worldwide. Video surveillance is a very popular one. Surveillance systems are not only for recording the observed visual information, but also extracting motion information and, more recently, to analyze suspicious behaviors in the scene [2]. One can visually track airplanes, vehicles, animals, micro-organisms or other moving objects, but detecting and tracking people is of great interest. For instance, vision-based people-counting applications can provide important information for public transport, traffic congestion, tourism, retail and security tasks. Tracking humans is also an important step for human-computer interaction (HCI) [3]. Video can also be processed to obtain the story, to group similar frames into shots, shots into scenes or to retrieve information of interest.

The objective of this overview paper is to explore the different approaches and provide comprehensive descriptions of different methods used for object detection and tracking. Our aim is to introduce recent advances in visual object tracking as well as identifying future trends. The remainder of the paper is structured as follows. Section 2 discusses the approaches related to object modeling. Section 3 presents motion detection including modeling of environments and shadow removal. Section 4 describes the different prediction methods. Section 5 reviews the work related to object tracking. Section 6 discusses the evaluation measures and datasets used for evaluation and comparison of object tracking methods. Finally, section 7 presents concluding remarks and future directions.

1.1 Problem Definition

Object tracking itself is the task of following one or more objects in a scene, from their first appearance to their exit [4]. An object may be anything of interest within the scene that can be detected, and depends on the requirements of the application. Given a sequence of image frames to trace a set of objects, which are

subimages, in each frame. In general, in a dynamic environment both background and object are allowed to vary. In principle, to solve this general unconstrained problem is hard. One can put a set of constraints to make this problem solvable. The more the constraints, the problem is easier to solve. Some of the constraints that generally imposed during object tracking are:

- Object motion is smooth with no abrupt changes
- No sudden changes in the background
- Gradual changes in the appearance of object
- Fixed camera
- Number and size of objects
- Limited amount of occlusion

Let denotes the image at time. Then, a video can be defined as the concatenation of images during different times, as following:

$$\upsilon = \{I_t : t = 1..\mathrm{T}\}$$

where T is the time frame when the video stop.

Figure 1 shows the general structure and representation of a video. In a color video, each frame consists of three components: R, G and B, while in a grayscale video, each frame has a single component.

There are two important facts which require attention for object tracking in video:

1. Video is a temporal sequence of image frames and video coding generally encode the temporal relationship. However, when image frames are regenerated, each frame exists independently of each other and their temporal relationships can be seen visually and are to be derived again, if needed.
2. Objects are embedded in the background and both are part of the image (frame), which is generally represented as an array of pixels. The spatial relationship, which groups pixels as object, are to be derived explicitly, may be in each frame.

Deviation of these spatio-temporal relationships forms the core of all object tracking algorithms.

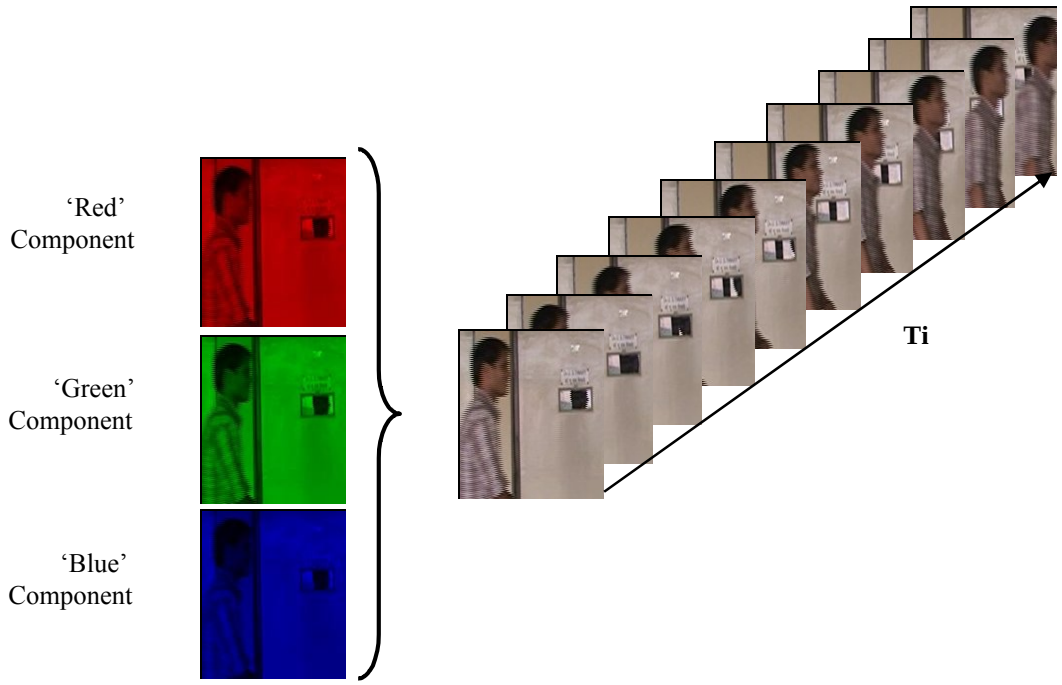


Figure 1: Video structure and representation.

1.2 Major Issues in Object Tracking

Many approaches have been proposed in the literature for object tracking. These approaches can be distinguished based on the way they handle the issues: i) segmentation algorithm is to extract moving objects in a video; ii) object representation for robust object tracking; iii) image features used to detect object in the feature space iv) handling of occlusion and v) the motion modeling. Some of the fundamental open problems in object tracking are abrupt object motion, noise in the image sequences, changes in scene illumination, changing appearance patterns of the object and the scene, object-to-object and object-to-scene occlusions, non-rigid object structures, camera motion and real time processing requirements. There are a number of issues involved in the development of a robust object tracking system, which needs to be understood.

Object Modeling is an important issue in visual object tracking. One of the major tasks of object modeling is to find an appropriate visual description that makes the object distinguished from other objects and background.

Changes in appearance and shape are issues that should also be considered during visual object tracking. The appearance of an object can vary as camera angle changes. Deformable objects such as human can change their shape and appearance during different video frame sequences. The appearance and shape can also change

due to perspective effect i.e. objects farther from the camera appears smaller than those near to the camera.

Handling ***illumination changes*** is also one of the challenging issues for visual object tracking. The appearance of an object can largely affected by illumination changes. An object may look different in indoor environment (artificial light) than outdoor environment (sun light). Even the time of day (morning, afternoon, evening) and weather conditions i.e. cloudy, sunny etc. can be the causes of illumination changes.

Shadows and reflections are also difficult to handle during object tracking. Some of the features such as motion, shape and background are more sensitive for a shadow on the ground which behaves and appears like the object that casts it. Same kind of problem can be caused by reflections of moving objects on smooth surfaces.

Occlusion is also very important issue for visual object tracking. Occlusion occurs either due to one object is occluded by another object or an object is occluded by some component of the background. During occlusion, an ambiguity occurs in the objects and their features. The tracking methods must be capable to resolve the individuality of the objects involved in the occlusion, before and after the occlusion takes place.

The issues mentioned above are significant to both single-object tracking and multi-object tracking. However, multi-object tracking also requires to resolve some other issues e.g. modeling the multiple objects. Tracking method should be able to distinguish different objects in order to keep them consistently labeled. Although during the last few years, there has been a substantial progress towards moving object detection and tracking. But tracking an object in an unconstraint, noisy and dynamic environment still makes this problem a central focus of research interest.

1.3 Typical Object Tracking Architecture

The visual object tracking field relies on three modules that interact with each other to perform robust object tracking.

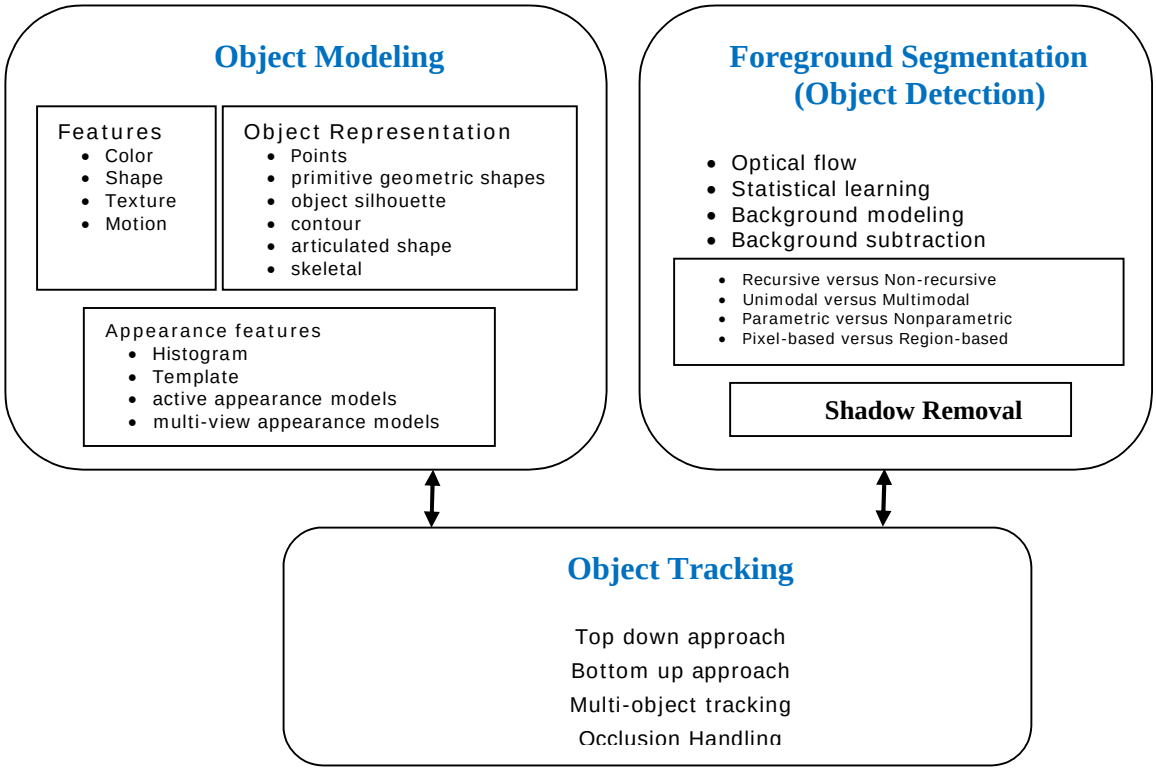


Figure 2: Functional architecture of visual object tracking,

2 Object Modeling

Object modeling plays a crucial role in visual tracking because it characterizes an object of interest. Selecting an effective object model plays a critical role in object tracking. Only the feature defined by the object model is used to maintain the estimate of the track. Object modeling therefore consists of two attributes: the representation of the object, which describes its span in the frame, and the features, which characterize it. Consequently, a poor choice of object model inevitably leads to poor tracking. The range of object representations encompasses various types of models and is application dependent. Some applications only require a simple model, while others require accurate and complex object models to achieve tracking.

2.1 Object Representation

Two important aspects that determine the performance of the tracking algorithms are object representation and object localization. Object representation refers to how

the object to be tracked is modeled and object localization deals with how the search of the corresponding object in the following frame is accomplished. From the object representation point of view amongst the wide variety of approaches adopted it can distinguish those that use a minimum amount of information extracted from the object, like color [5], intensity [6], feature points [7], spatialized color histograms [8]. Also, it can be used the integration of multiple number of features to have a better representation of the object [9]. There are also the approaches that use a very specific model of an object; this basis is useful when the goal is to track solid models. These approaches are based mostly on the contour, edges or a more detailed representation of an image curve using a parameterization like B-splines [10].

Object shape representations generally used for tracking are: points, primitive geometric shapes (e.g. rectangle, ellipse), object silhouette, contour, articulated shape and skeletal models [11] as shown in Figure 3.

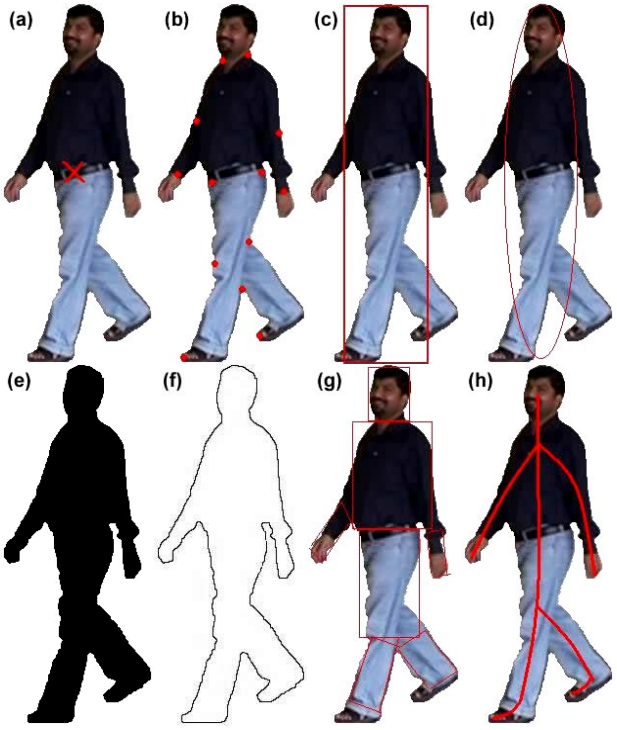


Figure 3: Object shape representations (a) point (b) multiple points c) primitive geometric shape (rectangular) d) primitive geometric shape (elliptical) e) silhouette f) contour (g) articulated shape (h) skeletal model.

Point Representation: In visual object tracking, the trivial shape is the point. An object is represented with a pixel location representing either some statistics on the object, such as the centroid, or a particular characteristic of interest. Point representation has been used in a plethora of applications due to its processing simplicity and the ease of point manipulation with complex algorithms [12]. For instance, it has been used for point tracking in radar imagery [13], distributed point tracking [14] or for Monte Carlo techniques where the number of samples prohibits heavy calculations [15], [16], [17]. Point tracking also alleviates the uncertainty regarding the position of the object of interest in the frame since it is based on a single point. It can be complemented with various order moments describing the distribution of the shape, such as the variance of pixels in the object of interest [18], [19]. Points have also been used to generate heuristics on some characteristics of the object. They are also used in the calculation of optical flow: due to the large number of vectors to estimate, only the point representation can be afforded [20] [21].

Primitive geometric shapes: The point representation of an object is a simple model. However, it does not grasp the entire dynamics of the object. For instance, rotation is not catered for with point representation. More advanced parametric shapes are, therefore, necessary to address these types of problems. The popular parametric shapes are primitive geometric shape such as rectangle, square, ellipse and circle. They are more appropriate for representing simple rigid objects. However using adaptive methods they can also be used for non-rigid objects. The rectangle representation is ubiquitous in geometric object tracking such as cars [22], [23] or in low-distortion object tracking such as people [24]. An adaptive square shape has been used for object representation in [25]. The ellipse offers the advantage of "rounding" the edges compared to the rectangle when the object does not have sharp edges [26]. In [8], [27], the author used an elliptical shapes to represent the moving object.

Articulated shape models: Articulated shapes are employed for tracking if different portions of the object of interest are to be described individually (*e.g.* legs, arms and head). This kind of representation is much suitable for a human body, which is an articulated object with head, hands, legs etc. These constituent parts should be related by a kinematic model. The constituent parts can be represented by any primitive geometric shape such as rectangles, circles and ellipses. Ramanan and Forsyth developed an articulated shape model to describe the body configuration and disambiguate overlapping tracks [28] In [29] the position of the different body limbs to analyze the behavior of people.

Skeletal models: In this representation a skeleton of object can be extracted to model both articulated and rigid objects. We can define the skeleton as a set of articulations within an object that describes the dependencies and defines constraints between the representations of the parts. In [30] the author utilized the skeletal model for automatic segmentation and recognition of continuous human activity.

Object silhouette: The silhouette is also called 'Blobs'. A blob is a dense, non-disjoint, binary mask that represents an object of interest. Blobs are of particular importance for pixel-wise processing. For instance, background subtraction provides blobs identifying the foreground or the moving objects in a scene [31], [32] [33].

Contour: In this representation the boundary of an object is defined as a contour. It provides a convenient non-parametric trade-off between an exhaustive

description of the object and storage requirements. Instead of storing the entire silhouette, contours only describe the edges enclosing the object. A non-rigid object shape can be better represented by these representations [34].

The appearance features of objects can also be characterized by a number of methods including probability densities of object appearance, templates and active appearance models [11].

Histogram approach is the most popular probability density estimates of the object appearance. The color histogram is relatively unaffected by pose change or motion, and so is also a reliable metric for matching after occlusion. However, one limitation of histograms is that they do not contain any position information. Two objects that have very similar color histograms may have dramatically different appearances due to the distribution of the colors. The color correlogram [35] is a variant of the color histogram, where geometric information is encoded as well as color information according to predefined geometric configurations.

Templates are formed using simple geometric shapes or silhouettes. A comprehensive description of the use of templates in computer vision can be found in [36]. Templates aim to represent objects with a set of predefined models. In that sense, templates can be categorized as semi-parametric representations. The predefined models are *a priori* non-parametric and can be of arbitrary form, providing single or multiple views of the object of interest. However, the matching of the model is performed by projection, distortion, scaling, etc., which are parametric transforms. One of the main tasks concerning templates is to maintain the set of models to minimize their number and maximize their relevance to the scene. First, if the appearance of the object is assumed to be static, the set of templates can be generated at initialization and updates are not necessary [37]. If the object changes appearance but is limited to a pre-defined range, the set of templates can be learnt off-line [38], thereby limiting its size. Another approach is on-line update and pruning of the set throughout time [39]. Templates are simple non-parametric representations to manipulate due to the restriction in the set of models and the parametrization of transforms used for matching. Nguyen et al. [40] performed a normalized correlation template tracking in the modulation domain. For each frame of the video sequence, they compute a multi-component AM-FM image model that characterizes the local texture structure of objects and backgrounds.

An *Active Appearance Models (AAMs)* contains a statistical model of the shape and grey level appearance of the object of interest [41]. They incorporate both shape and texture into their formulation; hence they enable us to track simultaneously the outline of an object as well as its appearance. It is therefore easy to use the parameters provided by an AAM tracker in other applications. Stegmann [42] demonstrated that AAMs can be successfully applied to perform object tracking. In his deterministic approach, the AAM search algorithm is applied successively to each frame.

2.2 Object Features

The object can be modeled by their shapes and appearances. Figure 4 illustrates that the object can be represented at different level of abstraction. At the low level the object can be represented simply by intensity value of its pixels. At the middle level it can be represented by some features like color, texture etc. At the highest level it can be represented by a global feature vector which can be boosted from many features. In general, a tracking framework exploits a global feature vector to measure the similarity between target and candidate object.

The major issue is finding an appropriate visual description for an object so that it can be uniquely define in the feature space and easily distinguished from others. The ideal feature for object tracking is an invariant of the object, i.e. at least robust to any type of transform, any change of illumination, any degradation. Some of the features such as color, shape, texture, and motion can be used to describe objects.

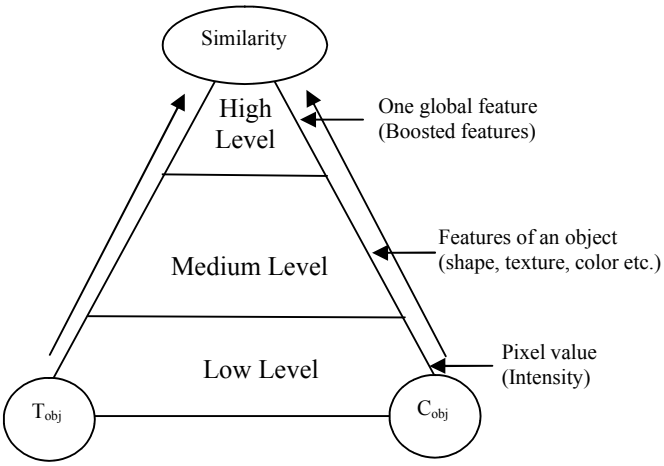


Figure 4: Object representation and matching in feature space (T_{obj} : Target object, C_{obj} : Candidate object).

Color Modeling: Color is most fundamental feature to describe an object. Due to its strong descriptive power, color is a good choice for representing an object [43]. RGB color space is usually used to represent images; however, the RGB color model is perceptually not a uniform color model. HSV is an approximately uniform color space and used intensively in literature. Hue, saturation and value are the three components of a HSV color space. In general, color spaces are sensitive to illumination change and noise. In [25], a single channel (hue) is considered in the color model.

Shape Modeling: The shape features are used as a powerful cue to detect object in video frame sequences. The shape of an object can be represented by a set of control points on the spline [44]. Edge is also used as feature where boundary of the objects is used to track the object. For more detail review on edge detection, the reader is referred to [45]. The advantage of edge feature over color feature is that edge is less sensitive to illumination changes.

Texture Modeling: Texture is also an important identifying characteristic of images. It is used to measure

the intensity variation of a surface and concerned with representing regular patterns in an image [4]. The texture representations can be classified into two classes: structural and statistical. Morphological operator and adjacency graph are two structural methods used to describe texture. Statistical methods include 1-D grey-level histograms, co-occurrence metrics, grey-level differences and multi-resolution filtering methods. As compared to color, texture features are also less sensitive to illumination changes.

Motion Modeling: Motion detection is vital part of the human vision system [44]. It is one of the functions of rod cells of our eyes. Optical flow is the most widespread depiction of motion. Optical flow represents motion as a displacement vectors which defines the movement of each pixel in a region between subsequent frames [46]. Horn and Schunk [46] computed displacement vectors using brightness constraint, which assumes brightness constancy of corresponding pixels in consecutive frames. Lucas-Kanade [47] proposed a method that computes optical flow more robustly over multiple scales using a pyramid scheme.

3 Foreground Segmentation

Detection of object to be tracked is the primary step in object tracking process. The object can be detected either once in the first frame or in every frame in the video. The goal of segmentation is to find out the semantically meaningful regions of an image and cluster the pixels belonging to these regions. It is very expensive to segment all static objects in an image. However, it is more practical to segment only the moving objects from video using spatio-temporal information in sequence of images (frames). A segmentation method should be generic and should not depend on other factors like color, shape and motion. Also, segmentation method should not be computationally intensive and should require less memory.

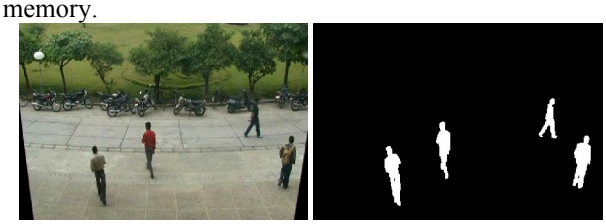


Figure 5: Foreground mask for an outdoor scene (Manual segmentation).

Foreground objects are defined as objects that are moving or involve in some activity. To track these objects, they have to be separated from background scene. A background scene is assumed as temporally stationary component of the video-frame such as roads, buildings and furniture. The Figure 5 shows an example of a foreground mask, where white and black colors represent the foreground and background pixels respectively.

Although a lot of studies have been conducted in recent years, the subject is still challenging. Some of the popular approaches proposed in the literature include background subtraction method, optical flow method and statistical learning method (non-parametric kernel density estimation). Algorithmic complexity is the major disadvantage of optical flow method. It requires higher time span than other methods. The non-parametric kernel density estimation method stores color values of multiple frames and estimates the contributions of a set of kernel functions using all of the data instead of iteratively updating background models at each frame [33]. The requirement of training samples and higher computational complexity make these methods infeasible for real time processing [48].

The following are some usually referred classifications discussed in the literature [49].

Recursive versus Non-recursive - Recursive techniques [31], [32], [50] use a single background model that is periodically updated. Non-recursive methods [51], [52], [53], [54] estimate a background model using statistical properties of previous frames by keeping a buffer. So non-recursive technique requires higher memory in comparison to recursive technique.

Unimodal versus Multimodal – In unimodal methods [31], a single modality is used to model the intensity of a pixel. On the other hand, multimodal methods [32] are used to represent the multi-modality of the underlying scene background. Although these methods cope with multi-modal distributions caused by shadows, waving tree branches, flickering monitor etc., but at the cost of higher complexity.

Parametric versus Nonparametric – Parametric methods [31], [32] require a tricky parameter initialization. The nonparametric models [33], [48] are more flexible and not require any assumptions about the underlying distributions. However, nonparametric models are memory and time consuming.

Pixel-based versus Region-based – Pixel-based methods [31], [32] use the statistical properties of individual pixel to model the background. While region based methods [33], [54] assume that foreground pixels have a tendency to appear in sets of connected points as a region.

3.1 Background Subtraction

The background subtraction method is one of the very simple and promising approaches for extracting moving objects from video sequences [55]. In background subtraction approach, we compare current frame with a reference frame known as background image. A significant difference indicates the presence of moving objects. However, if the reference frame is not modeled or updated adequately, this approach can be highly vulnerable to environmental conditions like illumination and structural background changes. So background modeling is one of the primary and challenging tasks for background subtraction. The background subtraction algorithm should be robust against environmental changes i.e. capable to handle changes in illumination conditions and able to ignore the movement of small background elements.

In recent years, several methods for performing background modeling and subtraction have been

proposed. Frame differencing [55] is a simple and easy way to extract moving object from a video sequence. In this approach, the image difference between consecutive frames is used and considerable difference in pixels value is considered as foreground region. However, Frame differencing methods suffer from fat boundary and thresholding problem. Another background modeling method is the use of temporal median filter, proposed by Lo and Velastin [51]. In this the median value of the pixels in the last 'n' frames is taken as the background model. In [52], the author extended this model for color images. Cucchiara et al. [53] proposed a mediod filtering approach, in which the mediod of the pixels can be computed from the buffer of image frames. However, this approach does not produce a measure of variance.

Wren et al. [31] proposed a method to model the background independently at each pixel location using a single Gaussian distribution. A recursive updation using a simple linear filter is used to estimate the Gaussian parameter. This technique is very simple and having fast implementation. However, it fails whenever some kind of variations occurs in the background.

Stauffer and Grimson [32], [50] proposed a method known as Gaussian Mixture Model (GMM), to handle multi-modal distributions using a mixture of several Gaussians. The GMM is the most representative approach and has been widely used [56]. The weight (w), mean (μ) and the standard deviation (σ) of a Gaussian component is updated recursively to imitate the new observations for pixel value. For the unmatched distributions the mean and variance remain unchanged but weights decrease exponentially. The matched components are updated by using a set of equations. These equations boost the confidence in the matched component by increasing w, decreasing σ, pushing μ towards the pixel value. A component is considered a matched component if the difference of mean and pixel's intensity value is less than a scaling factor (D) of a background component's standard deviation σ. A confidence metric (w/σ) is used to decide which components are parts of the background model. This is useful to select 'M' most confident guesses. M is the maximum number of modes one expects in the background probability distribution function. The first M components whose weight w is larger than a specified threshold become background model. Those pixels that don't match with any Gaussian components are treated as foreground pixels. The GMM can deal with multimodal distribution. However, the major disadvantages of GMM are that it is computationally intensive and require a tricky parameter optimization.

Elgammal et al. [33] exploited a nonparametric kernel density estimation to build a background PDF. The probability density estimation is performed using the recent historical samples without any assumption about background and foreground. The model is robust and has good model accuracy as compared to Gaussian mixture model in more complex scenes. However, the high computation cost, limits its scope.

Recently, a method based on texture is proposed to model the background and extract moving objects [57]. A binary pattern calculated around the pixel in a circular region is used to model each pixel. The binary pattern indicates whether the neighbouring pixel is smaller or larger than the central pixel. A modified local binary pattern (LBP) operator is used to extract features to make the method invariance to monotonic gray-scale change. However, the method can cause poor performance on flat image areas, where the intensity values of the neighbouring pixels are similar. Though this texture based method belongs to nonparametric methods, but it is fast due to simplification of LBP computation.

In [56], the author exhibits that the output of a background segmentation algorithm in the form of foreground segmentation masks can be significantly improved by applying post-processing techniques. This post-processing includes noise removal, morphological opening, closing operation, area thresholding etc.

3.2 Shadow Removal

There are many processes that are used to improve the performance of foreground segmentation. These processes include suppression of shadow, reflection and handling ghosts. Among these processes shadow suppression is most crucial task, which helps in improving detection accuracy and avoids analysis failure. After background subtraction, we get pixels correspond to objects as well as shadows. This is because shadow pixels are also detected different from the background and adjacent to object pixels and merge in a single blob as shown in Figure 6(b). Shadows occur when an object exists between a source of illumination and the surface on which it rests. These are natural phenomena and are easily perceptible to the human eye. However, shadows cause a lot of complications in various computer vision algorithms like object segmentation, object recognition, object tracking, scene understanding, etc. This is because the shadows tend to move in similar patterns and directions as an object being tracked, thus getting detected as a part of the object [58]. The shadow areas appear as surface features and corrupt the original object area, resulting in misclassification of object of interest and bias in estimation of object parameters. It is clearly depicted in Figure 6(c) that due to shadow the bounding box representation of the object becomes incorrect and contains a large portion of background. As a result any features computed for higher level analysis give an incorrect end results.

Shadows are mainly of two types, self shadow and cast shadow [59]. A 'self-shadow' is one which occurs on the object itself and is perceived as the darker regions of the body in the direction opposite to the direction of illumination. These are usually obscure and gradually change in intensity, with no definite boundaries. On the other hand, 'cast-shadow' is the dark region projected on the ground by occlusion of light due to the object. These tend to have hard, distinct shapes with sharp boundaries. Cast shadows are a major issue in object tracking and object recognition tasks. A number of approaches are proposed in literature to suppress cast shadow. A comparative evaluation of shadow detection methods is

given by Prati et al. in [58]. They proposed a two layer classification to highlight differences between different shadow removal algorithms. In the first layer, the approaches are classified as statistical and deterministic. In statistical methods, a probabilistic function is used to classify the shadow pixels whereas deterministic methods use an on/off decision process. The statistical approaches are further classified in parametric and non-parametric classes. The deterministic approaches are also further divided in model based and non-model based approaches. In model based techniques prior knowledge can be used to represent the model. On the other hand, non-model based methods use spectral and temporal properties to detect shadows. It is concluded that in case of noisy environment a statistical approach outperform as compared to deterministic model. It is also suggested that fewer assumptions should be defined to handle shadow problem in more generalized way.

Figure 6: Moving object segmentation for an outdoor scene a) Original frame b) foreground mask c) bounding box representation of object.

Hsies et al. [60] proposed a shadow elimination method based on statistical model using Gaussian shadow modeling. They used a coarse-to-fine shadow modeling approach. At the coarse stage, an orientation of the detected moving object mask is computed using central moment. Then shadow is detected by computing the rough boundaries between the cast shadow and the moving object, using difference of histogram (orientation, vertical) and silhouette features. A Gaussian shadow modeling is used to further refine the rough approximation of the shadow area. Parameters such as orientation, illumination and position are used for this purpose.

Early researches for shadow removal were typically focused on identifying dark areas on plain and flat surfaces. In [61], Nadimi et al. proposed physics based approach of shadow detection. A Gaussian mixture model is used for background modeling. They followed a multistage approach. At each stage the pixels are filtered out, which cannot be shadow pixels. The stages are as follows:

1. Initial shadow pixel reduction- In a training phase, the body color of surface that may come under shadow in the scene, is calculated. Only pixels having attenuated intensity than their background (in R, G, B) are considered as shadow candidates.
2. Blue ratio test - A blue ratio test exploits the fact that the illumination due to blue sky is responsible for outdoor shadows, so there is a higher ratio of blue.
3. Albedo ratio segmentation – An albedo ratio is used to extract regions of uniform reflectance. The albedo ratio is computed by combining two components. The first is the ratios of difference between two neighboring pixels and second is the ratios of differences between foreground and background pixels.
4. Ambient reflection correction – In this step foreground pixel values are subtracted from the background pixels, to suppress the effect of sky illumination.
5. Body color segmentation – In this step dichromatic reflection model is utilized to compute the true color of the object.
6. Verification – Finally verification is used to match the various surfaces with their expected body colors and determines which regions lie in shadow.

No spatial assumptions are considered in Nadimi's approach. However, the approach is supervised and mainly suited for outdoor situations.

The Dichromatic reflection model is also used in [59]. Initially a candidate shadow regions is identified using the hypothesis that shadow darken the surface on which it is cast upon. Then a verification stage is applied based on photometric invariant color features and geometric properties. Hue is used as the color invariance feature which is expected to be unchanged between shadows and object regions.

In [53], Cucchiara et al. proposed a general-purpose approach to extract the moving objects, ghosts, and shadows. They used a HSV color space and exploit the statistical assumptions that in a shadow region, the brightness and saturation properties are reduced while hue properties remain same. The ratio of reduction lies in the range α to β [53]. The first range (α) represents a maximum value for the darkening effect and depends on the intensity of light source, while the second range (β) is imposed to avoid detecting points that have been slightly altered by noise. The ghost object can be separated by analyzing the optical flow. As ghost object does not represent any motion, so they have an optical flow either zero or inconsistent.

A number of methods use the spectral property of shadow to indentify it [53], [59], [61]. However, the spectral properties fail to resolve the candidate shadow region accurately when the object body is darker than the background. In such cases geometrical properties can provide valuable information for shadow segmentation. In [59], [62], geometry properties such as shadow-background boundary and shadow-object boundary are used as an aid to verify the existence of shadows.

In [63], Leone and Distante exploited texture analysis for shadow detection with the assumption that textural properties remain same in shadow-regions. A Gabor function is used to perform texture analysis. Although the texture based method are simple, however, these methods mostly good for the identification of weak shadows, indicating its use for indoor environments. Also these methods are computationally intensive.

4 Prediction Methods

For tracking objects in a video, we have to find the position of object's instances in two consecutive frames. One of the brute force methods is to exploit the matching technique on the whole image of every incoming frame. But this puts an overhead of exhaustive search. In general, the tracking algorithm assumes that in a few consecutive frames the trajectory of object does not change abruptly. Therefore, to increase the efficiency of the algorithm, the matching technique is not exploited in whole image frame. Rather the reference template is matched in a search space, which will be somewhere in the surrounding of the region where last time the object was detected. Predicting possible location of candidate object in each frame will also help in improving tracking accuracy and minimizing the search space. The robustness of the system to handle abrupt motion and occlusion is also influenced by this prediction. The better the prediction of the object location is, the search space become smaller. Thus the accurate algorithm can additionally speed up the whole tracking process. There are three common approaches to predict an object's position.

- Motion Model
- Kalman Filter
- Particle Filter

The simplest type of predictor is the *motion model*. Motion model exploits past observation to predict the next position [64]. A simple motion model can be formulated as,

$$l(t+1) = l(t) + v(t)$$

where $l(t)$, $l(t+1)$ represents the current location and the predicted location at t and $t+1$ time step respectively, and $v(t)$ is the velocity at t time and is defined as,

$$v(t) = l(t) - l(t-1)$$

Acceleration can also be used in motion model.

Predictive filtering can also be used as one of tracking approach. The *Kalman filter* is a linear predictive filter and widely explored in the vision community for tracking [65]. It is proved to be good to predict the state, in the presence of noise. Kalman filtering consists of two steps, prediction and correction. The prediction step estimates the process state at the next time step, using previous state variables. While the correction step incorporates the new observations into the system to update the object's state. A detailed explanation of the mathematical equations is discussed in [65]. However, Kalman filter restricted its application to only linear dynamic and measurements models with additive Gaussian noise. To handle the non linear relationships, an extension is proposed such as Extended Kalman filter (EKF) [65].

In [66] the author proposed an algorithm which adaptively predicts possible coordinate transform parameters for the next frame and selects them as the initial searching point when looking for the real transform parameters. An adaptive Kalman filter is used, but instead of directly filtering the values of transform parameters, the Kalman filter is applied on the changing rate of those parameters to effectively predict their future values.

Another major problem with Kalman filter is that it can model only a single hypothesis. Due to the unimodal Gaussian assumption, it is not feasible to represent multiple hypotheses simultaneously using the Kalman filter. This limitation can be overcome by using *particle filtering*, which is based on Monte Carlo integration methods [67]. In particle filtering, a set of random samples with associated weights are used to compute the current density of the state (which can be location, size, speed, boundary etc.). These samples and weights are also used to compute the new density.

5 Object Tracking

The goal of an object tracker is to create the trajectory of an object over time by locating its position in every frame of the video. Tracking can also be defined as detecting the object and maintain correspondence between object instances across every frame of the video. The features of a good tracking algorithm are as follows;

1. The tracking algorithm should detect all the objects that enter or moved in the scene.
2. The tracking algorithm should differentiate between multiple objects that are present in the scene at the same time.
3. To monitor and extract the trajectory of all objects the unique label assigned to each object must be maintained for all the tracked objects.
4. The motion or lack of motion of the object should not lead to change of object label.
5. The tracking algorithm should handle occlusion and exposure without object labels changing.

There are two distinct methodologies to approach the tracking problem, top down (forward tracking) and bottom-up (back-tracking). Top down methods are goal oriented and find the positions of the object in the current frame using a hypothesis generated at the start of the tracking based on parametric representation of the target. On the other hand, in a bottom-up approach, the moving objects are detected in every frame and then a correspondence is established with the objects those were detected in the previous frame. A representative of top down approach is many model based and template matching approach [27], [25]. While a blob based tracking represents a bottom up approaches [31].

5.1 Top down Approach

The top down approaches often rely on external input to initialize the tracking process. These tracking methods use different object characteristics, such as color, texture, shape and motion. One of the popular method in this category is mean-shift based tracking [8], [27], [25]. Mean shift tracker exploits the concept of non-parametric density gradient estimator that iteratively executed within the local search kernels [68]. It uses the color histogram

to model object probability density and moves the object region in the largest gradient direction.

In [25], Bradski proposed an adapted version of mean-shift called CAMShift (Continuously Adaptive Mean-shift). A histogram based on known hue value in color image sequences is used to track the head and face movement. Mean shift algorithm is used with adaptive region sizing step. The kernel having simple step function is applied to a skin probability map. The mean location i.e. centroid at each iteration with the search window is computed as zero and first order moment. Due to the consideration of a single channel (hue), the algorithm is supposed to consume less CPU time. However, the algorithm may fail to track objects having multiple hue value or objects, where hue value is not sufficient to discriminate the object from background.

Comaniciu and Meer [27] proposed a kernel based tracking algorithm. A weighted color histogram is used as feature to represent the target object in an ellipse. The weighted histogram is computed using Epanechnilov kernel profile which assigns smaller weights to pixels farther from center. The author used Mean shift to find the location of target model in the current frame. Bhattacharyya coefficient has been used as a measure of comparability between the target object and the candidate object. The location of the target object in previous frame is used as a starting point for Mean shift procedure in current frame. The Mean shift procedure maximizes the value of similarity measure i.e. Bhattacharyya coefficient iteratively. Although the kernel based method is computationally simple, however, the method fails as soon as the color distribution of object becomes similar with any other region in image frame. This is because color histogram does not contain any position information. Two objects that have very similar color histograms may have dramatically different appearances due to the distribution of the colors. For example, one person may be wearing a white shirt and black pants whilst a second is wearing a black shirt and white pants. Whilst these people may have quite distinct appearances, they would have very similar histograms.

One of the problems of Mean shift is that it is designed to find local maxima for tracking objects. As a result Mean shift tracker may fail in case of large target movement between two consecutive frames. In [69], the author proposed a multibandwidth procedure to help conventional MS tracker to reach the global mode of the density function using any staring points.

There are also a number of literatures on the problem based on fragments of object tracking. In [70], Adam et al. presented fragment-based tracking which accounts for partial occlusions. It uses a computationally and memory expensive technique called integral histograms. Through exhaustively searching, there is no formal framework by which we can selectively use certain fragments. Some significant improvements in Mean shift tracking are suggested in [71], where a fast target updation scheme using foreground separation to tackle appearance change is proposed. To improve robustness, the object to be tracked is divided into more than one fragment. Whenever a new frame is extracted from video sequence, candidate model is built for each such fragment and Mean shift is used to find a pair of target object. An enhanced kernel based object tracking system is developed that uses background information and edge. Color marginal histogram is exploited to tackle the scale change problem. The coordinates of the winning fragment of the object location using the most confident estimate is used to obtain the object position. However, there is no self driven ways by which we can get the most confident estimate of the object. Moreover, the intrinsic feature selection would result in a tracking drift or decreasing the weights of some target blobs. In [72], a fusion scheme has been proposed to fuse multiple spatially distributed fragments. Under the fusion scheme, a mean shift type algorithm which allows efficient target tracking with very low computational overhead. However, the weight of each fragment in occlusion will result in draft. A Mean shift based multiple model tracking algorithm is proposed in [73]. The author exploits several connected regions to incorporate spatial information into object representation. Multiple models are used to adapt changes in object appearance during the tracking process. Switching between multiple models has been done using Bayes probabilistic rule.

In [74], the author proposed a multi-kernel approach to handle fast motion area and improves the convergence problem by integrating two likelihood terms. In [75], Fang et al. proposed an efficient and robust fragments-based multiple kernels tracking algorithm. A feature, which is based on the separation of the foreground/background likelihood function, is used for tracking.

Recently, many methods exploited multiple features to improve reliability and tracking performance [76], [77]. In [76], the authors integrate the shape-texture and color features in the Mean shift tracking framework. The shape-texture feature is represented by an orientation histogram. Ning et at. [77] extended the mean-shift tracking algorithm by combing a LBP texture feature with color histogram features. In [78], geometric features are used for real-time vehicle tracking. Texture patterns give the spatial structure of an object. However, texture feature become ineffective, where objects having large smooth regions. Hu et al. [79] extracted three histograms from each person, one each for the head, torso and legs, to not only allow for matching based on color, but also on distribution of color. Shen et al. [80] extend the use of statistical learning algorithms for object localization and tracking. In contrast to building a template from a single frame, a probabilistic kernel-based SVM is used to represent object model from a large amount of samples.

In designing an appearance model, the crucial properties that a tracker needs to meet are robustness and adaptability to changes in target appearance (e.g.,pose, illumination). Recently, many tracking methods are developed to achieve these goals by incorporating an adaptive appearance model. Ross et al. [81], proposed a Incremental Visual Tracker (IVT) and represented a target as a low-dimensional subspace that captures the principal components of possible appearance variations,

where the subspace is updated adaptively using the image patches tracked in the previous frames. Unlike many non-adaptive approaches that employ fixed appearance template models, this method alleviates the burden of constructing a target model prior to tracking with a large number of expensive offline data, and tends to yield higher tracking accuracies. However, the model is restricted to characterizing only texture-rich objects. A robust tracking algorithm based on the adaptive pixel-wise appearance model is proposed in [82]. Intensity value of each pixel in appearance model is modeled by a mixture of Gaussian density whose parameters are updated using sequential kernel density approximation.

5.2 Bottom up Approach

The bottom up approach covers those methods which uses the background modeling and subtraction approach to extract foreground objects and then track the objects by establishing a unique correspondence with the previously detected targets over time. In [31], the author proposed a system named as 'Pfinder', for detecting and tracking human body. The background is modeled separately for each pixel location. It uses a Gaussian probability density function in the YUV space on the last '*n*' pixel's value. In each new frame the statistics is updated using running Gaussian average. Multiple blobs are used to model the person's various body parts. Each blob is represented by spatial information, color component and the corresponding Gaussian distributions. In each new frame the scene and the person model is dynamically changing. A Kalman filter is used to predict spatial distribution for current frame. The log likelihood method is used to resolve the class membership i.e. decision about the pixel assigning to the background scene or one of the blobs. Then iterative morphological operations are used to produce a single connected region and the statistical models for the blob and scene texture model are updated. The person body parts like head, hands and feet are labeled using a 2-D contour shape analysis. The skin color is used to initialize hand and face blobs. The method shows good results in indoor environment, however, its success in outdoor scenes is not explored much.

Zhixu Zhao et al. [83], proposed a texture based multi-target tracking algorithm. A local binary patterns (LBP) is used as texture descriptor. A single Gaussian model is used for background modeling to perform foreground segmentation. A Kalman filter is used as a motion predictor. A LBP histogram distance is used to distinguish blob in case of occlusion.

In [84], the author proposed an integrated framework for object detection and tracking. A Support Vector Regression (SVR) is used to model background. The background model is updated online over time. A confidence coefficient computed using shape, color and motion information is used to improve target-to-target correspondences over time.

5.3 Tracking in Wavelet Domain

Most of the algorithms discussed in previous sub-sections are unable to track objects in the presence of noise, variations in illumination, appearance and camera jittering, as most of these algorithms working in spatial domain use features which are sensitive to these variations [85]. In recent years, the wavelet feature based techniques have gained popularity in object tracking. It provides a rich and robust representation of an object [86], due to the following characteristics:

- Wavelet transform provides powerful insight into an image's spatial and frequency characteristics [87], [88].
- Provides an efficient framework for representation and storage of images at multiple levels [89].
- Provides noise resistant ability [90].
- High frequency sub-bands of wavelet transform represent the edge information [91].
- Wavelets are also a core technology in the next generation compression methods [92].

One of the features of discrete wavelet transform (DWT) is that the spatial information is retained even after decomposition of an image into four different frequency coefficients. In this decomposition, the high frequency sub images (horizontal coefficient, vertical coefficient and diagonal coefficient) contain the detailed information. In [93], a rule based method is proposed to track object between video images sequences. First the moving object is isolate from background in each wavelet transformed frame. Then, a feature is computed based on the positions, the size, the grayscale distribution and presence of textures of objects. However, the method is computationally expensive and unable to deal with occlusion. A wavelet based vehicle tracking system was proposed in [94]. A frame difference analysis of two consecutive frames has been used to extract the moving object. After the removal of shadow from extraction object, a wavelet-based neural network is used recognized the moving vehicles. The centroid and wavelet features difference measures are used to track the identified objects. They highlight that decomposing an image at lower level using WT reduces the computational complexity. However, still the recognition step acquires a major computational cost. In [95] the author used highest energy coefficients of Gabor wavelet transform to model the object in the current frame and 2D mesh structure around the feature points to achieve global placement of the feature point. The 2D golden section algorithm is used to find the object in the next frame.

A real-time multiple object tracking algorithm is proposed in [96]. In this algorithm, wavelet coefficients is not used as object features, rather the original frame is only preprocessed using a 2-level discrete wavelet transform to suppress the fake background motions. The difference image of successive frames is computed using the approximation band of the wavelet transform. Then, the object is identified using the concept of connected components in the difference image. The identified

objects are then represented by a bounding box in the original approximation image. This bounding box representation is used to compute some color and spatial features. In the successive frames these features are then used to track the objects. Chang et al. [97] proposed a tracking system based on discrete wavelet transform to track a human body. A CCD camera is used, which is mounted on a rotary platform for tracking moving objects. The background subtraction method is used for object detection. The YIQ (Y: luminance, I & Q values jointly describe the hue and Saturation) color coordinate system is used as a feature for tracking. The second level Haar DWT is used to pre-process the images for reducing computation overhead. However only single human object can be tracked and also need a background model (background don't having object).

In object tracking we require any object feature which remains invariant by translation and rotation of the object. A real wavelet is used in most of the papers discussed above. However, one of the major problems with real wavelet transform is that it suffers from shift-sensitivity [98]. In [85], [99] authors used an undecimated wavelet packet transform (UWPT) to overcome the problem of shift sensitivity. Amiri et al. [99] proposed an object tracking algorithm based on a block matching method in using Undecimated wavelet packet tree (UWPT). For block matching in wavelet domain they used the motion vector for each pixel of reference block. The method for finding best match among FVs of the reference block and FVs of the search window is called "Dispersion of Minimums (DOM)". A object tracking algorithm for crowded scenes based on pixel features in the wavelet domain and a adaptive search window updating mechanism based on texture analysis have been proposed in [85]. An adaptive feature vector generation and block matching algorithm in the UWPT domain is used for tracking objects in crowded scenes in presence of occlusion and noise. In addition, an inter-frame texture analysis scheme is used to update the search window location for the successive frames. However, the UWPT expansion is redundant and computationally intensive.

A Daubechies complex wavelet transform [100] can be a better solution which is also approximately shift-invariant. Not many researchers have explored complex wavelet transform (CxWT) application to tracking problems. Recently [88] has shown the applicability of CxWT to denoising and deblurring. Recently an object tracking method based on Daubechies complex wavelet transform domain is proposed [101]. A complex wavelet domain based structural similarity index is used, which is simultaneously insensitive to small luminance change, contrast change and spatial translation. The reference object in the initial frame is modeled by a feature vector in terms of the coefficients of Daubechies complex wavelet transform. They illustrated that the proposed algorithm has good performance even in noisy video with significant variations in object's pose and illumination. Figure 7 shows that the Daubechies complex wavelet transform based method gives accurate results even in the presence of noise.

Figure 7: Tracking in noisy video (Gaussian Noise with) using CWT tracker [101].

5.4 Multi-Object Tracking and Occlusion Handling

Mutli-object tracking algorithms should be able to establish unique correspondences between objects in each frame of a video. Tracking multiple objects of the same class implies that the tracking method must be able to discriminate objects, especially when the objects are similar in appearance. Occlusion may be of different types: self occlusion, inter object occlusion, object to background occlusion [102] as shown in figure 8. Self occlusion takes place when an object occludes itself e.g. the face of a person can be occluded by its hand. On the other hand, inter object occlusion occurs due the partial or full overlapping of more than one objects. While a background occlusion occurs when the tracked object is occluded by some component of the background.

Split may occur due to merged objects or because of errors in the segmentation method. An error in the split may mislead the tracker. A good multi-object tracking method should be able to detect changing numbers of objects in the scene, adding and removing objects when appropriate and also able to handle both occlusion and split events.

Although object tracking has been explored a lot but very little work has been done to resolve the issues of multiple object tracking. Kalman filtering is an efficient solution to track multiple objects [103]. However, mistakes become more frequent and are difficult to correct as the number of objects increases. The problem can be solved using particle filtering by exploiting the multiple hypotheses [104]. In [105], the authors formulate the multi-object tracking as a Bayesian network inference problem and explore this approach to track multiple players. In [106], the author proposed a probabilistic framework based on HMM to describe a

multiple object trajectory tracking. The framework was able to track unknown number of multiple objects. The association problem has been represented as a bipartite graph in [107]. A method was proposed to maintain hypotheses for multiple associations. They also resolved the problem of objects entering and exiting, and handled the error due to merging and splitting objects. However, particle filter-based tracking algorithms having not enough samples that are statistically significant modes, faced difficulty to track multiple objects. They are only capable to handle partial short duration occlusion.

Figure 8: The different types of occlusion states: a) Self Occlusion b) Inter object occlusion c) Background occlusion.

Okuma et al. [108] incorporates an Adaboost learning machine into a single particle filter to differentiate multiple targets. The problem with a joint state is the high computational cost since generic importance sampling becomes less efficient without exploiting the intrinsic correlation among the targets as the number of targets increases. Cai et al. [109] extend the work [108] by proposing an individual mean-shift embedded particle filter for each hockey player, while the association problem is solved by an extra nearest neighbor (NN) algorithm. By moving sampled particles to stabilized positions using mean-shift, the posterior probability is better approximated with fewer particles. The overhead is the mean-shift procedure for each particle.

Senior et al. [110] used an appearance template having a RGB color model with associated probability mask, to represent each pixel of the object. The probability mask represents the probability of the corresponding pixel belonging to that template. The appearance model is continuously updated. The background is statistically modeled and a background subtraction method is then used to extract the foreground regions. A distance matrix is used to establish the correspondence between each foreground region with one of the active objects. Using learned appearance model and the predicted object positions, the occluded foreground region is separated in corresponding objects. However, in case of similar colored/textured objects, the approach becomes vulnerable to distinguish the objects.

In [111], Rad and Jamzad discussed three criteria to predict and detect occlusion of vehicles in a highway. The first criterion is based on examining the trajectory of each vehicle. Occlusion can be predicted, if the centroid of two foreground regions is too closed to each other. Second and third criteria are based on size of foreground region. If a drastic change is found in the size between two consecutive frames then there may be possibility of a merge or split event. A bounding contour of the motion mask is used to resolve the occlusion state.

In [112], the author proposed a multi-object tracking system to track pedestrians using a combined input from RGB and thermal cameras. The motion in the scene is modeled using a particle filter based Bayesian framework. Benezeth et al. [113] proposed a vision-based system for human detection and tracking in indoor environment using a static camera. The moving objects are extracted using a background subtraction approach exploiting a single Gaussian model. Multiple cascades of boosted classifiers based on Haar-like filters are used to know the nature of various objects

In [114], a two stage nonlinear feature voting strategy based method is used for tracking multiple moving objects. The first stage i.e. object voting is used to match two objects. Whereas the second stage i.e. corresponding voting is used to resolve confusion in matching in case of multiple matches. A rule based approach with no appearance model is used to track objects in presence of noise, occlusion and split problem. The method is able to track multiple objects in presence of heavy occlusion and split using the plausibility rules. However, it fails when the objects suddenly disappears or change its direction.

Lao and Zheng [115] discussed special problem of multi-target tracking, where a group of targets are highly correlated, usually demonstrating a common motion pattern with individual variations. They proposed a algorithm which explore the correlations among the targets in a statistical online fashion and embeds both the correlation and the most recent observations into sampling to improve the searching efficiency.

6 Evaluation Measures and Data Sets

6.1 Evaluation Measures

Evaluation of the performance of moving object detection and tracking algorithm is one of major task to validate the correctness and robustness of the tracking algorithm. In March 2000, the first initiative was taken in form of a workshop known as *Performance Evaluation of Tracking and Surveillance* (PETS). Since then, several such PETS workshops have been organized, including most recently a workshop in which the focus was on event level task. The consortium on *CLassification of Events, Activities, and Relationships* (CLEAR) established in 2006 as a collaboration of the European CHIL (Computers in the Human Interaction Loop) project and US National Institute of Standards and Technology (NIST). The goal was to establish a universal evaluation framework for tracking and other related tasks.

The evaluation of different object detection and tracking methods can be performed in two way i.e.

qualitatively and *quantitatively*. Qualitative evaluation approaches are performed on visual interpretation, by looking at processed image yield by the algorithm. On the other hand, quantitative evolution requires a numeric comparison of computed results with ground truth data. Due to the necessity of computing a valid "ground truth" data, the quantitative (experimental) evaluation of object detection and tracking algorithms are highly challenging. Figure 9 illustrates an example of qualitative evaluation and it is obvious that tracker_1 has better tracking accuracy.

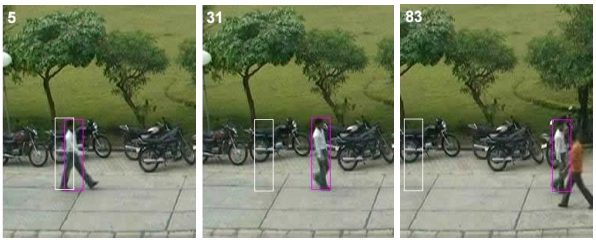


Figure 9: Qualitative result of tracking [101]: tracker_1 (indicated with magenta box), tracker_2 (represented by white box).

In [110] the author proposed a number of metrics for evaluating performance of tracking. These measures includes #track false positives, #track false negatives, average position error, average area error, average detection lag, and average track incompleteness. However, the dependency on input data is the major limitation of these measures. Tracker Detection Rate (TDR), False Alarm Rate(FAR), Object Tracking Error (OTE), Track Fragmentation (TF) and Occlusion Success Rate (OSR) are the evaluation measures proposed in [116] to finds the correspondence between ground truths and tracked objects to compute true positive and false positive matches. The measures defined in [116] are as follows:

$$\text{Tracker Detection Rate } (\text{TRDR}) = \frac{\text{Total True Positives}}{\text{Total Number of Ground Truth Points}}$$

$$\text{False Alarm Rate } (\text{FAR}) = \frac{\text{Total False Positives}}{\text{Total True Positives } + \text{ Total False Positives}}$$

$$\text{Track Detection Rate } (\text{TDR}) = \frac{\text{Number of true positives for tracked object}}{\text{Total number of ground truth points for object}}$$

$$\text{Occlusion Success Rate } (\text{OSR}) = \frac{\text{Number of successful dynamic occlusions}}{\text{Total number of number of dynamic occlusions}}$$

$$\text{Tracking Success Rate } (\text{TSR}) = \frac{\text{Number of non} - \text{fragmented tracked objects}}{\text{Total number of number of ground truth objects}}$$

$$\text{Object Tracking Error } (\text{OTE}) = \frac{1}{N_{rg}} \sum_{\exists i\ g(t_i) \Lambda r(t_i)} \sqrt{(xg_i - xr_i)^2 + (yg_i - yr_i)^2}$$

where N_{rg} is the number of frames used for tracking and $(xg_i,xr_i)(yg_i,yr_i)$ is the location of the ground truth and result track at frame *i* respectively. Ground truth points that reside inside the bounding box are referred to as a true positive. While a ground truth point that is not located within the bounding box is referred as false negative.

Similar to OTE, in [117] the author measure the performance of tracking algorithm by finding the localization error, which can be calculated as $SQRT((Gx_i-X_i)^2 + (Gy_i-Y_i)^2)$, where (Gx,Gy) is the centroid representing ground truth and (X_i,Y_i) is the centroid of computed tracked object. The graph is depicted in figure 10.

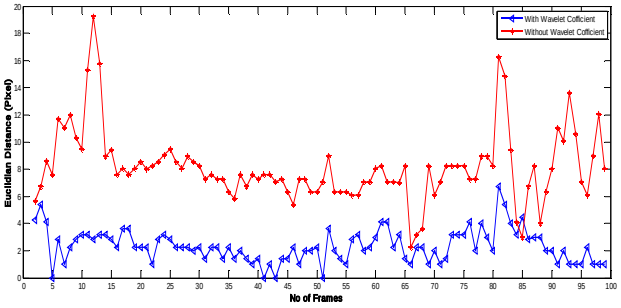


Figure 10: Resulting graph used in [117] to show error of object location using the Euclidian distance (in number of pixels) in each frame.

In [118] the author presented an evaluation method that determines the threshold from the distance matrix between the centroid of the bounding box for the ground truths and the result of the tracking algorithm. False Positive Track Error, False Negative Track Error, Average Area Error, and Task Incompleteness Factor measures are computed using the correspondence between the result of the tracking algorithm and the ground truth data.

Yin et al. [119] presented a new set of metrics to assess different aspects of performance of motion tracking. They proposed statistical metrics, such as Track matching Error (TME), Closeness of Tracks (CT) and Track Completeness (TC) that indicate the accuracy of estimating the position, the spatial and temporal extent of the objects respectively. They also discussed metrics, such as Correct Detection Track (CDT), False Alarm Track (FAT) and Track Detection Failure (TDF) to show the overview of the algorithm performance.

Smith et al. [104] attempted to describe an objective procedure to measure multiple object tracker performance. They have evaluate the configuration and identification performance of multi-object tracking systems by measuring the configuration errors as False Positive (FP), False Negative (FN), Multiple Trackers(MT), Multiple Objects(MO) and identification errors as Falsely Identified Tracker(FIT), Falsely Identified Object (FIO), Tracker Purity(TP), Object Purity(OP).

In [120] the author proposed two intuitive and general metrics to allow for objective comparison of tracker characteristics, focusing on their precision in estimating object locations, their accuracy in recognizing object configurations and their ability to consistently label objects over time. They defined two very intuitive metrics as multiple object tracking precision (MOTP) and the multiple object tracking accuracy (MOTA). The MOTP describe the total error in estimated position for matched object-hypothesis pairs over all frames, averaged by the total number of matches made. The MOTA compute the accuracy in terms of the number of missed detects, false positives, and switches in the system output track for a given reference ground truth track.

The *Multiple Object Tracking Accuracy (MOTA)* was defined as:

$$MOTA = \frac{1 - \sum_{t=1}^{N_{frames}} (c_m(m_t) + c_f(fp_t) + c_s(ID - SWITCHES_t))}{\sum_{t=1}^{N_{frames}} N_G^{(t)}}$$

where, m_t is the number of misses, fp_t is the number of false positives, and $ID - SWITCHES_t$ is the number of ID mismatches in frame t considering the mapping in frame $t-1$. Therefore, during tracking, if there was a track split or merge, one would still consider the contribution of the new track but penalized it by counting it as an $ID - SWITCHES$. The values used for the weighting functions in this evaluation were $c_m = c_f = 1$ and $c_s = \log_{10}$. ID-SWITCH count is started from 1 because of the log function.

The *Multiple Object Tracking Precision (MOTP)* was defined as:

$$MOTP = \frac{\sum_{i=1}^{N_{mapped}} \sum_{t=1}^{N_{frames}^{(t)}} \left[\frac{\left| G_i^{(t)} \cap D_i^{(t)} \right|}{\left| G_i^{(t)} \cup D_i^{(t)} \right|} \right]}{\sum_{t=1}^{N_{frames}} N_{mapped}^{(t)}}$$

where N_{mapped} refers to the mapped system output objects over an entire reference track taking into account splits and merges and N_{mapped}^{t} mapped refers to the number of mapped objects in the t_{th} frame.

A comprehensive overview of object detection and tracking evaluation measures has been given in [121]. The author systematically address the challenges of object detection and tracking through a common evaluation framework that permits a meaningful objective comparison of detection and tracking techniques used for face, text and vehicle objects in video. They discussed the need of a large development and evaluation corpus that can be used to support machine learning approaches and statistically differentiate differences in system performance. A comprehensive description of evaluation measures is presented, which permit system performance differences to be discerned via a minimal number of measures. Overall they discussed the necessary infrastructure (source video, task definitions, metrics, ground truth, and scoring tools) to perform formal evaluations of face, text, and vehicle detection and tracking tasks.

In order to provide a quantitative perspective about the quality of foreground detection two of the measures are very popular among the researcher [122]. These measures are, the false negative rate (FNR) and false positive rate (FPR) and defined as

$$\text{FNR} = \frac{\text{the number of foreground pixels wrongly classified}}{\text{the number of foreground pixels in the ground truth}}$$

$$\text{FPR} = \frac{\text{the number of background pixels wrongly classified}}{\text{the number of background pixels in the ground truth}}$$

In [123] it was discussed that FNR, FPR measures not accurate enough, when averaging the measures over various environments. The author proposed a new formulation to evaluate the foreground segmentation. The new similarity measure was:

$$\text{S(A, B)} = \frac{\text{A} \cap \text{B}}{\text{A} \cup \text{B}}$$

where A and B represent detected region and ground truth region respectively. The value of S(A, B) lie between 0 (lest similarity) to 1. This measure integrates the FNR and FPR into a single measure. However, it is a nonlinear measure. Another measures used to quantify the performance of moving object segmentation are recall and precision [124].

$$\text{Recall} = \frac{\text{Number of foreground pixels correctly identified by the algorithm}}{\text{Number of foreground pixels in ground truth}}$$

$$\text{Precision} = \frac{\text{Number of foreground pixels correctly identified by the algorithm}}{\text{Number of foreground pixels detected by the algorithm}}$$

The value of recall and precision fall within the range of 0 and 1. A good background algorithm should attain as high a recall value as possible without sacrificing precision. In order to quantify the performance of the shadow detection method, two measures are proposed in [125], which are shadow detection rate (η) and shadow discrimination (ξ) and defined as:

$$\eta = \frac{\text{TP}_S}{\text{TP}_S + \text{FN}_S}; \quad \xi = \frac{\overline{\text{TP}_F}}{\text{TP}_F + \text{FN}_F}$$

Where TP represent the number of true positives, the $\overline{\text{TP}_F}$ is the number of ground truth points of the foreground objects minus the number of points detected as shadows, but belonging to foreground objects. The subscript S stands for shadow and F for foreground.

6.2 Data Sets

Data set is one of key components of any system. Evaluating the algorithm against a standard dataset is one of the challenging tasks in object tracking. In the recent years a number of common data set is available by different communities. The Performance Evaluation of Tracking and Surveillance (PETS) series of workshops was unique in the tracking community in that a common data set was used by all of the workshop participants. In the following section we are giving details of these data sets.

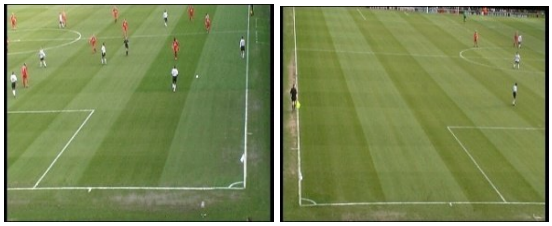
- PETS'2000: Outdoor people and vehicle tracking (single camera)
  ftp://ftp.pets.rdg.ac.uk/pub/PETS2000/

- PETS'2001: Outdoor people and vehicle tracking (two synchronised views; includes omnidirectional and moving camera) (annotation available)
  ftp://ftp.pets.rdg.ac.uk/pub/PETS2001
  http://www.cvg.cs.rdg.ac.uk/PETS2001/pets2001-dataset.html

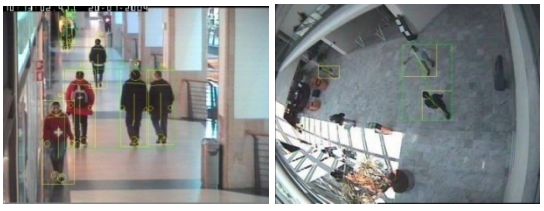
- PETS'2002: Indoor people tracking (and counting) and hand posture classification data (annotated).
  ftp://ftp.pets.rdg.ac.uk/pub/PETS2002
  http://www.cvg.cs.rdg.ac.uk/PETS2002/pets2002-db.html

- PETS ICVS'2003: Annotation of a smart meeting (annotation available). Includes facial expressions, gaze and gesture/action.
  ftp://ftp.pets.rdg.ac.uk/pub/PETS-ICVS
  http://www.cvg.cs.rdg.ac.uk/PETS-ICVS/pets-icvs-db.html

- VS PETS 2003: Outdoor people tracking - football data (two views). The datasets consists of football players moving around a pitch.
  ftp://ftp.pets.rdg.ac.uk/pub/VS-PETS
  http://www.cvg.cs.rdg.ac.uk/VSPETS/vspets-db.html



- PETS ECCV'2004: CAVIAR people scenarios
  http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/



- PETS'2006: Multi-sensor sequences containing left-luggage scenarios with increasing scene complexity.
  ftp://ftp.pets.rdg.ac.uk/pub/PETS2006/

- PETS'2007: The datasets are multisensor sequences containing the following 3 scenarios, with increasing scene complexity: 1. loitering, 2. attended luggage removal (theft), 3. Unattended luggage.
  ftp://ftp.pets.rdg.ac.uk/pub/PETS2007/
  http://pets2007.net/

- PETS'2009: The datasets are multisensor sequences containing different crowd activities.
  http://www.cvg.rdg.ac.uk/PETS2009/a.html

- SPEVI (Audiovisual people dataset): This is a dataset for uni-modal and multi-modal (audio and visual) people detection tracking. The dataset consists of three sequences recorded in different scenarios with a video camera and two microphones.
  http://www.elec.qmul.ac.uk/staffinfo/andrea/spevi.html

7 Conclusion and Future Directions

Over the past one decade moving object detection and tracking has continued to be a booming area of research. The demands of potential mass-market applications in surveillance, human computer interaction and video retrieval makes this area favorite among the researcher. Increased activity in this research area has been driven by both academia and industry.

In this overview paper, we have discussed some of the core concepts used in object tracking and present a comprehensive survey of efforts in the past to address this problem. In the recent year, an important issue that has been got attention is the integration of knowledge (contextual information) in the design of tracking methods. In [126], an unsupervised data mining approach

has been integrated to reduce the uncertainty in the tracking process. A set of auxiliary objects are found during the process, which gives extra information to help the tracking process. The occlusion can be resolved by exploiting the motion correlations among the auxiliary object and the target. A multi-object tracking framework exploiting the scene contextual information (target births, spatially persistent clutter) as feedback was proposed in [127]. A GMM is used to model birth and clutter data.

Knowledge of multi modality can also help in tracking and specially occlusion handling, because each modality may introduce new information that compensate the weaknesses of other. In [128] the author proposed a framework for object tracking using joint statistical characteristics of the audio-video data. The two modalities were fused at semantic level to help the tracking task.

Although during the last few years there has been a substantial progress towards object detection and tracking. But tracking an object in an unconstraint, noisy and dynamic environments are still makes this problem a central focus of research interest. Developing a background model robust and efficient to environmental changes is still a challenging task. Exploitation of prior and contextual knowledge in tracking is still in its initial phase. Tracking objects in noisy and compressed video data is also required a serious attention. Most of the past researches have focused independently on object detection and object tracking. Thus there is great demand of developing a robust and efficient approach that can incorporate the task of object detection, tracking and analysis in a single framework. Although in the recent years a number of benchmark data set (e.g. PETS data set) and evaluation measures are proposed by different communities but still the development of a common benchmark data and evaluation measures, which can cover all kind of scenario, is a critical requirement for object detection and tracking. Future work can also focus on development of parallel version of tracking algorithm so that one can utilize the processing capability of network resources.

References

[1] T. Acharya and A. K. Ray (2005). *Image processing*: *principles and applications*, New Jersey: Wiley-Interscience.

[2] M. Shah, O. Javed and K. Shafique (2007). Automated visual surveillance in realistic scenarios. *IEEE MultiMedia*, vol.14, no.1, pp.30-39.

[3] W. M. Hu, T. N. Tan, L. Wang and S. Maybank (2004). A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on Systems, Man and Cybernetics*, vol.34, no.3, pp. 334-352.

[4] D. A. Forsyth and J. Ponce. (2003) *Computer vision: a modern approach*, Prentice Hall.

[5] P. Perez, C. Hue, J. Vermaak and M. Gagnet (2002). Color-Based Probabilistic Tracking. *In Proceedings of ECCV*, pp. 661–675.

[6] K. Pahlavan and J.O. Eklundh (1992). A head-eye system- analysis and design. *CVGIP: Image Understanding*, vol. 56, pp. 41–56.

[7] P. Tissainayagam and D. Suter (2005). Object tracking in image sequences using point features. *Pattern Recognition*, vol. 38, pp. 105–113.

[8] D. Comaniciu, V. Ramesh and P. Meer (2000). Real-time tracking of non-rigid objects using mean shift. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp.142-149.

[9] D. Serby, E. K. Meier and L.V. Gool (2004). Probabilistic object tracking using multiple features. *In Proceedings of International Conference on Pattern Recognition*, pp. 184–187.

[10] R. Cipolla and M. Yamamoto (1990). Stereoscopic tracking of bodies in motion. *Image and vision computing*, vol. 8, no. 1, pp. 85–90.

[11] A. Yilmaz, O. Javed and M. Shah (2006). Object tracking: a survey. *ACM Journal of Computing Surveys*, vol. 38, no.4, Article 13.

[12] C. Veenman, M. Reinders, E. Backer (2001). Resolving motion correspondence for densely moving points. *IEEE Trans. Patt. Analy. Mach. Intell.*, vol. 23, no. 1, pp. 54–72.

[13] L. M. Novak (1981). Optimal target designation techniques. *IEEE Transactions on Aerospace and Electronic Systems*, vol. 17, no.5, pp. 676–684.

[14] N. K. Kanhere and S. T. Birchfield (2008). Real-time incremental segmentation and tracking of vehicles at low camera angles using stable features. *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 1, pp.148–160.

[15] D. Angelova and L. Mihaylova (2008). Extended object tracking using monte carlo methods. *IEEE Transactions on Signal Processing*, vol. 56, no.2, pp.825–832.

[16] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P. J. Nordlund (2002). Particle filters for positioning, navigation, and tracking. *IEEE Transactions on Signal Processing*, vol. 50, no.2, pp.425–437.

[17] C. Kwok, D. Fox, and M. Meila (2004). Real-time particle filters. *Proceedings of IEEE*, vol. 92, no.3, pp.469–484.

[18] Y. Chen, G. Liang, K. K. Lee, and Y. Xu (2007). Abnormal behavior detection by multi-svm-based bayesian network. *In Proceedings of the International Conference on Information Acquisition*, pp. 298–303.

[19] X. Wu, Y. Ou, H. Qian, and Y. Xu (2005). A detection system for human abnormal behavior. *In Proceedings of the International Conference on Intelligent Robots and Systems*, pp. 1204–1208.

[20] D. Kragic and H. I. Christensen (2000). Tracking techniques for visual servoing tasks. *In Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1663–1669.

[21] G. Unal, H. Krim, and A. Yezzi (2005). Fast incorporation of optical flow into active polygons. *IEEE Transactions on Image Processing*, vol. 14, no. 6, pp. 745–759.

[22] J. Melo, A. Naftel, A. Bernardino, and J. Santos-Victor (2006). Detection and classification of highway lanes using vehicle motion trajectories. *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no.2, pp.188–200.

[23] C. Shu-Ching, S. Mei-Ling, S. Peeta, and Z. Chengcui (2003). Learning-based spatio-temporal vehicle tracking and indexing for transportation multimedia database systems. *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, no. 3, pp.154–167.

[24] C. Yang, R. Duraiswami, and L. Davis (2005). Fast multiple object tracking via a hierarchical particle filter. *In Proceedings of the IEEE International Conference on Computer Vision*, vol. 1, pp. 212–219.

[25] G. Bradski (1998). Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journa*l, vol.2, no.2, pp.1-15.

[26] C. Chang, R. Ansari, and A. Khokhar (2005). Multiple object tracking with kernel particle filter. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 566–573.

[27] D. Comaniciu, V. Ramesh and P. Meer (2003). Kernel-based object tracking. *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol.25, no.5, pp.564-575.

[28] D. Ramanan and D. A. Forsyth (2003). Finding and tracking people from the bottom up. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 467–474.

[29] I. Haritaoglu, D. Harwood, and L. S. Davis (2000). W4: Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no.8, pp.809–830.

[30] A. Ali and J. Aggrawal (2001). Segmentation and recognition of continuous human activity. *In IEEE Workshop on Detection and Recognition of Events in Video*. Pp. 28–35.

[31] C. Wren, A. Azarbayejani, T. Darrell and A.P. Pentland (1997). Pfinder: Real time tracking of the human body. *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol.19, no.7, pp.780-785.

[32] C. Stauffer and W.E.L. Grimson (1999). Adaptive background mixture models for real-time tracking. *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pp.246-252.

[33] A. Elgammal, R. Duraiswami, D. Harwood and L. S. Davis (2002). Background and foreground modeling using non-parametric kernel density estimation for visual surveillance. *Proceedings of the IEEE*, pp.1151-1163.

[34] A. Yilmaz, X. Li and M. Shah (2004). Contour based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE Trans. Patt. Analy. Mach. Intell.* vol.26, no. 11, pp. 1531–1536.

[35] A. Rao, R. Srihari, and Z. Zhang (2000). Geometric histogram: a distribution of geometric configuration of color subsets. *in SPIE: Internet Imaging*. vol. 3964, pp. 91-101.

[36] R. Brunelli (2009). *Template matching techniques in computer vision: theory and practice*. Wiley.

[37] R. Bastos and J. M. S. Dias (2005). Fully automated texture tracking based on natural features extraction and template matching. *In Proceedings of the 2005 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, pp.180–183.

[38] F. Jurie and M. Dhome (2001). A simple and efficient template matching algorithm. *In Proceedings of the IEEE International Conference on Computer Vision*, vol. 2, pp. 544–549.

[39] L. J. Latecki and R. Miezianko (2006). Object tracking with dynamic template update and occlusion detection. *In Proceedings of the IEEE International Conference on Pattern Recognition*, vol. 1, pp. 556–560.

[40] C.T. Nguyen, J.P. Havlicek, M. Yeary (2007). Modulation domain template tracking. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8.

[41] T.F. Cootes, G. J. Edwards, and C. J. Taylor (1998). Active appearance models. *ECCV*, vol. 2, pp. 484–498.

[42] M. B. Stegmann (2001). Object tracking using active appearance models. *in Proc. 10th Danish Conference on Pattern Recognition and Image Analysis*. vol. 1, pp. 54–60.

[43] R. Gonzalez and R. Woods (2002). *Digital Image Processing*, 2nd edition, Prentice Hall.

[44] M. Sonka, V. Hlavac and R. Boyle (2008). *Image Processing, Analysis and Machine Vision*, 3rd edition, Singapore: Thomson Asia Pvt. Ltd.

[45] D. Ziou and S. Tabbone (1998). Edge detection techniques - an overview. *International Journal of Pattern Recognition and Image Analysis*, vol. 8, pp. 537–559.

[46] B.K.P. Horn and B.G. Schunk (1981). Determining optical flow. *Artificial Intelligence*, vol.17, no.1, pp.185-203.

[47] B.D. Lucas and T. Kanade (1981). An iterative image registration technique with an application to stereo vision. *Proc. of the Workshop on Image Understanding*, pp.674-679, 1981.

[48] A. Tavakkoli, M. Nicolescu, G. Bebis and M. Nicolescu (2009). Non-parametric statistical background modeling for efficient foreground region detection. *Machine Vision and Applications, Springer*, vol. 20, no.6, pp.395-409.

[49] L. Maddalena and A. Petrosino (2008). A self-organizing approach to background subtraction for visual surveillance applications. *IEEE Transactions on Image Processing*, vol.17, no.7, pp.1168-1177.

[50] C. Stauffer and W.E.L. Grimson (2000). Learning patterns of activity using real time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.22, no.8, pp.747-757.

[51] B.P.L. Lo and S.A. Velastin (2001). Automatic congestion detection system for underground

platforms. *Proc. Int'l Symp. Intelligent Multimedia, Video, and Speech Processing*, pp.158-161.

[52] S. Calderara, R. Melli, A. Prati and R. Cucchiara (2006). Reliable background suppression for complex scenes. *ACM international workshop on Video surveillance and sensor networks*, pp.211-214.

[53] R. Cucchiara, M. Piccardi and A. Prati (2003). Detecting moving objects, ghosts, and shadows in video streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.25, no.10, pp.1337-1342.

[54] K. Toyama, J. Krumm, B. Brumitt and B. Meyersv (1999). Wallflower: principles and practice of background maintenance. *Proc. 7th IEEE Conf. Computer Vision*, vol.1, pp.255-261.

[55] M. Piccardi (2004). Background subtraction techniques: a review. *Proc. IEEE International Conference on Systems, Man and Cybernetics*, vol.4, pp.3099- 3104.

[56] D. H. Parks and S. S. Fels (2008). Evaluation of background subtraction algorithms with post-processing. *Proc. IEEE Int'l Conf. Advanced Video and Signal-based Surveillance*, pp.192-199.

[57] M. Heikkila and M. Pietikainen (2006). A texture-based method for modeling the background and detecting moving objects. *IEEE Transactions Pattern Analysis Machine Intelligence*, vol.28, no.4, pp.657-662.

[58] A. Prati, I. Mikic, M.M. Trivedi and R. Cucchiara (2003). Detecting moving shadows: algorithms and evaluation. *IEEE Transactions. Pattern Analysis Machine Intelligence*, vol.25, no.6, pp.918-923.

[59] E. Salvador, A. Cavallaro and T. Ebrahimi (2004). Cast shadow segmentation using invariant color features. *Computer Vision and Image Understanding*, vol.95, no.3, pp.238-259, 2004.

[60] J.W. Hsieh, W.F. Hu, C.J. Chang and Y.S. Chen (2003). Shadow elimination for effective moving object detection by Gaussian shadow modeling. *Image and Vision. Computing*, vol.21, no.6, pp.505-516.

[61] S. Nadimi and B. Bhanu (2004). Physical models for moving shadow and object detection in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.26, no.8, pp.1079-1087.

[62] Liu Zhi Fang, Wang Yun Qiong and You Zhi Sheng (2008). A method to segment moving vehicle cast shadow based on Wavelet transform. *Pattern Recognition Letters, Elsevier*, vol.29, no.16, pp.2182-2188.

[63] A. Leone, C. Distante, and F. Buccolieri (2007). Shadow detection for moving objects based on texture analysis. *Pattern Recognition, Elsevier*, vol.40, pp.1222-1233.

[64] T. Broida and R. Chellappa (1986). Estimation of object motion parameters from noisy images. *IEEE Transactions. Pattern Analysis Machine Intelligence*, vol.8, no.1, pp.90-99.

[65] G. Welsh and G. Bishop (1995). An introduction to the kalman filter. Technical Report TR95-041, University of North Carolina, Chapel Hill, NC, 1995.

[66] Jiyan Pan, Bo Hu, and Jian Q. Zhang (2006). An efficient object tracking algorithm with adaptive prediction of initial searching point. *Proc of the IEEE Pacific-Rim Symposium on Image and Video Technology (PSIVT'06), Lecture Notes in Computer Science*, vol. 4319, pp. 1113-1122.

[67] L. Mihaylova, P. Brasnett, N. Canagarajah and D. Bull (2007). Object tracking by particle filtering techniques in video sequences. *Advances and Challenges in Multisensor Data and Information Processing*, vol. 8, pp.260-268.

[68] D. Comaniciu and P. Meer (2002). Mean shift: a robust approach toward feature space analysis. *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol.24, no.5, pp.603-619.

[69] A. Dargazany, A. Soleimani and A. Ahmadyfard (2010). Multi-bandwidth kernel-based object tracking. *Journal of Advances in Artificial Intelligence, Hindawi Publication*, Article ID.175603.

[70] A. Adam, E. Rivlin, I. Shimshoni (2006). Robust fragments-based tracking using the integral histogram. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 798–805.

[71] J. Jeyakar, R. V. Babu, and K.R. Ramakrishnan (2008). Robust object tracking with background-weighted local kernels. *Computer Vision and Image Understanding, Elsevier*, vol.112, no.3, pp.296-309.

[72] F. L. Wang, S. Y. Yu, J. Yang (2009). Robust and efficient fragments-based tracking using mean shift. *International Journal of Electronics and Communications*, pp. 1-10, 2009.

[73] M. Lucena, J. M. Fuertes, N. Blanca, Manuel J and M. Jimenez (2010). Tracking people in video sequences using multiple model. *Multimedia Tools and Applications*, Springer, vol.49, no.2, pp.371-403.

[74] F. Porikli and O. Tuzel (2005). Multi-kernel object tracking. *Proc. IEEE Int'l Conf. on Multimedia and Expo*., pp.1234-1237.

[75] J. Fang, J. Yang, H. Liu (2011). Efficient and robust fragments-based multiple kernels track. *International Journal of Electronics and Communications, Elsevier*, vol. 65, pp. 915-923.

[76] J.Q. Wang and Y.S. Yagi (2008). Integrating color and shape-texture features for adaptive real-time object tracking. *IEEE Transactions on Image Processing*, vol.17, no.2, pp.235-240.

[77] J. Ning, L. Zhang, D. Zhang and C. Wu (2009). Robust object tracking using joint color-texture histogram. *International Journal of Pattern Recognition and Artificial Intelligence*, vol.23, no.7, pp.1245-1263.

[78] F. Deboeverie, K. Teelen, P. Veelaert and W. Philips (2009). Vehicle tracking using geometric features. Advanced Concepts for Intelligent Vision Systems, LNCS, vol.5807, pp.506-515.

[79] M. Hu, W. Hu, and T. Tan (2004). Tracking people through occlusions. *Proceedings of the 17th International Conference on Pattern Recognition*, pp.724-727.

[80] C. Shen, J. Kim and H. Wang (2010). Generalized kernel-based visual tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, vol.20, no.1, pp.119-130.

[81] D. Ross, J. Lim, R. Lin and M. Yang (2008) Incremental learning for robust visual tracking. *International Journal of Computer Vision, Springer*, vol.77, no.1, pp.125-141.

[82] B. Zhang, W. Tian and Z. Jin (2008). Robust appearance-guided particle filter for object tracking with occlusion analysis. *International Journal on Electronics and Communications*, Elsevier, vol.62, no.1, pp.24-32.

[83] Z. Zhao, S. Yu, X. Wu, C. Wang and Y. Xu (2009). A Multi-target Tracking Algorithm using Texture for Real-time Surveillance. *Proc. IEEE Int'l Conf. on Robotics and Biomimetics*, pp. 2150-2155.

[84] J. X. Wang, G. Bebis and M. Nicolescu (2009). Improving target detection by coupling it with tracking. *Machine Vision and Applications, Springer*, vol.20, no.4, pp.205-223.

[85] M. Khansari, H. R. Rabiee, M. Asadi and M. Ghanbari (2008). Object tracking in crowded video scenes based on the Undecimated Wavelet features and texture analysis. *EURASIP Journal on Advances in Signal Processing*, Article ID. 243534.

[86] Y. Tang (2008). Status of Pattern Recognition with Wavelet Analysis. *International Journal, Frontiers of Computer Science, Springer*, vol.2, no.3, pp.268-294.

[87] N. G. Kingsbury and J. F. A. Magarey (1997). Wavelet transforms in image processing. *Proc. First European Conference on Signal Analysis and Prediction*, pp.23-34.

[88] D. Clonda, J.M. Lina and B. Goulard (2004). Complex daubechies wavelets: properties and statistical image modeling. *Signal Processing (Elsevier)*, vol.84, pp.1-23, 2004.

[89] S. Mallat (1989). A theory of multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.11, pp.674-693.

[90] Y. Xu, J.B. Weaver, D.M. Healy and J. Lu (1994). Wavelet transform domain filters: a spatially selective noise filtration technique. *IEEE Transactions on Image Processing*, vol. 3, pp.747-758.

[91] J. K. Romberg, H. Choi and R. G. Baraniuk (2001). Multiscale Edge Grammars for Complex Wavelet Transforms. *Proc. IEEE Int. Conf. on Image Processing*, pp.614-617.

[92] Z. Xiong, K. Ramchandran, M. T. Orchard and Y. Q. Zhang (1999). Comparative Study of DCT and Wavelet Based Image Coding. *IEEE Transactions on Circuits and Systems for Video Technology*, vol.9, pp.692-695.

[93] Y. Wang, F. John F. Doherty, Robert E. Van Duck (2000). Moving object tracking in video. *Proceedings of 29th IEEE Int'l Conference on Applied Imagery Pattern Recognition Workshop*, pp. 95-101.

[94] J. B. Kim, C. W. Lee, K. M. Lee, T. S. Yun, and H. J. Kim (2001). Wavelet-based vehicle tracking vehicle for automatic traffic surveillance. *Proceedings of IEEE Region 10 International Conference on Electrical and Electronic Technology*, TENCON, vol.1, pp.313-316.

[95] C. He, Y. F. Zheng and S.C. Ahalt (2002). Object tracking using the gabor wavelet transform and the golden section algorithm. *IEEE Transactions on Multimedia*, vol. 4, no. 4, pp. 528–538.

[96] F.H. Cheng and Y. L. Chen (2006). Real time multiple objects tracking and identification based on discrete wavelet transform. *Pattern Recognition*, vol. 39, no. 6, pp. 1126–1139.

[97] Shyang-Li Chang, Chen-Chien Hsu, Tsung-Chi Lu, Ti-Ho Wang (2007). Human body tracking based on discrete wavelet transform. *Proceedings of the 2007 WSEAS International Conference on Circuits, Systems, Signal and Telecommunications,* pp:113–122.

[98] I.W. Selesnick, R.G. Baraniuk and N. Kingsbury (2005). The dual-tree complex wavelet transform. *IEEE Signal Processing Magazine*, pp. 123-151.

[99] M. Amiri, H. R. Rabiee, F. Behazin and M. Khansari (2003). A New Wavelet Domain Block Matching Algorithm for Real-Time Object Tracking. *Proceedings International Conference on Image Processing*, vol. 3.

[100] J. M. Lina and M. Mayrand (1995). Complex Daubechies Wavelets. *Applied and Computational Harmonic Analysis,* vol. 2, pp. 219-229.

[101] A. S. Jalal and V. Singh (2011). Robust object tracking under appearance change conditions based on Daubechies complex wavelet transform. *International Journal of Multimedia Intelligence and Security, Inderscience*, vol. 2, no. 3, pp. 252-268.

[102] P. F. Gabriel, J. G. Verly, J. H. Piater and A. Genon (2003). The state of the art in multiple object tracking under occlusion in video sequences. *Proc. of the Advanced Concepts for Intelligent Vision Systems*, pp.166-173.

[103] A. Mittal and L. Davis (2003). M2tracker: a multi-view approach to segmenting and tracking people in a cluttered scene. *International Journal of Computer Vision, Springer*, vol.51, no.3, pp.189-203.

[104] K. Smith, D. Gatica-Perez and J.-M. Odobez (2005). Using particles to track varying numbers of interacting people. *Proc. Int'l Conf. on Computer Vision and Pattern Recognition*, pp.962-969.

[105] P. Nillius, J. Sullivan and S. Carlsson (2006). Multi-target tracking - linking identities using bayesian network inference. *Proc. Int'l Conf. on Computer Vision and Pattern Recognition,* pp.2187-2194.

[106] M. Han, W. Xu, H. Tao and Y. Gong (2007). Multi-object trajectory tracking. *Machine Vision and Applications, Springer*, vol.18, no.3, pp.221-232.

[107] S.W. Joo and R. Chellappa (2007). Multiple-hypothesis approach for multi-object visual tracking. *IEEE Transactions on Image Processing*, vol.16, pp.2849-2854.

[108] K. Okuma, A. Taleghani, D. Freitas, J.J. Little, D.G. Lowe (2004). A boosted particle filter: multitarget detection and tracking. *Proc. European Conf. on Computer Vision*, pp. 28–39.

[109] Y. Cai, N. De Freitas, J.J. Little (2006). Robust visual tracking for multiple targets. *Proc. European Conf. on Computer Vision*, pp. 107–118.

[110] A. Senior, A. Hampapur, Y.-L. Tian, L. Brown, S. Pankanti and R. Bolle (2006). Appearance models for occlusion handling. *Journal of Image and Vision Computing*, Elsevier, vol.24, no.11, pp.1233-1243.

[111] R. Rad and M. Jamzad (2005). Real time classification and tracking of multiple vehicles in highways. *Pattern Recognition Letter*, Elsevier, vol.26, no.10, pp.1597-1607.

[112] A. Leykin and R. Hammoud (2010). Pedestrian tracking by fusion of thermal-visible surveillance videos. *Machine Vision and Applications, Springer*, vol.21, pp.587-595.

[113] Y. Benezeth, B. Emile, H. Laurent and C. Rosenberger (2010). Vision-based system for human detection and tracking in indoor environment. *Special Issue on People Detection and Tracking of the International Journal of Social Robotics, Springer*, vol.2, no.1, pp.41-52.

[114] A. Amer (2005). Voting-based Simultaneous Tracking of Multiple Video Objects. *IEEE Transactions on Circuits and Systems for Video Technology*, vol.15, pp.1448-1462.

[115] Y. Lao, Y. F. Zheng (2011). Tracking highly correlated targets through statistical multiplexing. *Image and Vision Computing*. vol. 29, no.12, pp. 803-817.

[116] J. Black, T. Ellis, and P. Rosin (2003). A novel method for video tracking performance evaluation. *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pp. 125–132.

[117] A. S. Jalal, U. S. Tiwary (2009). A robust object tracking method for noisy video using rough entropy in wavelet domain. *Proceedings of the International Conference Intelligent Human Computer Interaction, Springer India*, ISBN 978818489203, pp. 113-121.

[118] S. Muller-Schneiders, T. Jager, H. S. Loos, and W. Niem (2005). Performance evaluation of a real time video surveillance systems. *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pp. 137–143.

[119] F. Yin, D. Makris, S. A. Velastin (2007). Performance evaluation of object tracking algorithms. *Proc. of the 10th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*.

[120] Keni Bernardin and Rainer Stiefelhagen (2008). Evaluating multiple object tracking performance: the CLEAR MOT Metrics. *EURASIP Journal on Image and Video Processing*, Article ID 246309.

[121] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, J. Zhang (2009). Framework for performance evaluation of face, test, and vehicle detection and tracking in video: data, metrics and protocol. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp.319-336.

[122] M. Heikkilä, M. Pietikäinen (2006). A texture-based method for modeling the background and detecting moving objects. *IEEE Trans. Pattern Anal. Machine Intelligence*, vol. 28, no. 4, pp.657-662.

[123] L. Li, W. Huang, I. Gu, Q. Tian (2004). Statistical modeling of complex background for foreground object detection. *IEEE Transactions on Image Processing*, vol. 13, no. 11, pp. 1459-1472.

[124] S.C.S. Cheung, C. Kamath (2005). Robust techniques for background subtraction in urban traffic video. *EURASIP Journal on Applied Signal Processing*, vol.14, pp. 2330–2340.

[125] A. Prati, I. Mikic, M.M. Trivedi, and R. Cucchiara (2003). Detecting moving shadows: algorithms and evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 918-923.

[126] Ming Yang, Ying Wu and Gang Hua (2008). Context-aware visual tracking. *IEEE Transaction On Pattern Analysis And Machine Intelligence*, vol. 31,no.7, pp. 1195-1209.

[127] E. Maggio and A. Cavallaro (2009). Learning scene context for multiple object tracking. *IEEE Transactions on Image Processing*, vol. 18, no. 8, pp. 1873-1884.

[128] M. J. Beal, N. Jojic and H. Attias (2003). A graphical model for audiovisual object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 828-836.