OBJECT TRACKING IN A VIDEO

A thesis

Submitted in fulfillment of the requirement for the Degree of

Master of Engineering in Computer Science and Engineering

Of

Jadavpur University

By

Subhashis Pal

Registration No.: 129004 of 2014-2015

Examination Roll No.: M4CSE1617

Under the Guidance of

Prof. Jamuna Kanta Sing

Department of Computer Science and Engineering

Jadavpur University, Kolkata-700032

India

2016

FACULTY OF COMPUTER SCIENCE AND ENGINEERING
JADAVPUR UNIVERSITY

Certificate of Recommendation

This is to certify that the dissertation entitled "OBJECT TRACKING IN A VIDEO" has been carried out by Subhashis Pal (University Registration No.: 129004 of 2014-2015 Examination Roll No.: M4CSE1617) under my guidance and supervision and be accepted in partial fulfillment of the requirement for the Degree of Master of Engineering in Computer Science and Engineering. The research results presented in the thesis have not been included in any other paper submitted for the award of any degree in any other University or Institute.

……

Prof. Jamuna Kanta Sing (Thesis Supervisor)
Department of Computer Science and Engineering
Jadavpur University, Kolkata-32

Countersigned

……

Prof. Debesh Kumar Das
Head, Department of Computer Science and Engineering,
Jadavpur University, Kolkata-32.

……

Prof. Sivaji Bandyopadhyay
Dean, Faculty of Engineering and Technology,
Jadavpur University, Kolkata-32.

FACULTY OF COMPUTER SCIENCE AND ENGINEERING

JADAVPUR UNIVERSITY

Certificate of Approval

This is to certify that the thesis entitled "OBJECT TRACKING IN A VIDEO" is a bona-fide record of work carried out by Subhashis Pal in fulfillment of the requirements for the award of the degree of Master of Engineering in Computer Science and Engineering in the Department of Computer Science and Engineering, Jadavpur University. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the thesis only for the purpose for which it has been submitted.

……..

Signature of Examiner 1

Date:

……..

Signature of Examiner 2

Date:

FACULTY OF COMPUTER SCIENCE AND ENGINEERING
JADAVPUR UNIVERSITY

Declaration of Originality and Compliance of Academic Ethics

I hereby declare that this thesis entitled "OBJECT TRACKING IN A VIDEO" contains literature survey and original research work by the undersigned candidate, as part of his Degree of Master of Engineering in Computer Science and Engineering.

All information have been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: Subhashis Pal
Registration No: 129004 of 2014-2015
Exam Roll No.: M4CSE1617
Thesis Title: Object Tracking in a Video

…..……..

Signature with Date

Acknowledgement

Any accomplishment requires the effort of many people and this work is no difference. I would like to express my deep gratitude and indebtedness to my teacher and guide Prof. Jamuna Kanta Sing, Department of Computer Science and Engineering, Jadavpur University, for his directive instructions, constant encouragement throughout the course of this work. His valuable guidance and encouragement have really led the path to the successful completion of this project. I would like to thank Prof. Debesh Kumar Das, Head of The Department of Computer Science & Engineering for providing me with the necessary facilities for carrying out the thesis.

Last but certainly not least; I would like to thank my parents and brother for their constant encouragement and patience during the course of studies. Needless to say, without all the above help and support the writing and production of this thesis would not have been possible.

…….……..

Subhashis Pal

Registration No: 129004 of 2014-2015

Exam Roll No: M4CSE1617

Department of Computer Science & Engineering

Jadavpur University

List of Figures

Contents

Chapter-1

Introduction

1.1 Overview

An ‘Object’ can be defined as “anything which can be seen or touched”. In computer vision terminology, “an object is considered as a continuous closed area in an image which is distinct from its surroundings”. Basically object is an area of utmost importance which needs to be observed. And ‘Tracking’ means “to observe or plot the movement instrumentally”, and in computer vision terminology tracking can be defined as “any means by which simultaneously localizing multiple objects and maintaining their identities”. Hence the term ‘Object Tracking’ means to constantly observe the motion of an object on a path or direction or both.

Object tracking has been gaining a lot of interest and attention from many researchers in the image processing field due to its academic and commercial value, for the last few decades. Object tracking is nothing but constantly observing the motion or direction of any object, which is of interest in consecutive frames in a video. The video can be online or offline in nature. In the online method the video which is been recorded by some optical means such as camera, by magnetic means such as MRI scans, electromagnetic means such as radar and the object tracking is done on that live video. In the offline method the video has been pre-recorded and stored in some storage device such as hard-disk, memory card and CD or DVD disc and there after object tracking is done on that video for object analysis.

Object tracking is done both by software and hardware. The software based object tracking is done on the pre-recorded video, which highlight the subject to be track. Multi-object tracking is done through software based object tracking. The hardware based tracking is somewhat different from software based tracking, in the hardware based tracking the hardware such as camera has to constantly focus and move with the subject i.e. rotating camera on axis to get the object into the frame.

Video is actually a sequence of images taken over a period of time. To perceive motion in video, the video has to be recorded in frame rate so that our eye cannot distinguish between two consecutive frames. This frame rate has to be more than our eye can distinguish images between two frames. Generally this is more than 12 frames per seconds which makes movement in videos looks better, to get a smoother video the speed is kept between 24-30 fps. More frame rate can store more information which is use in high speed videography. More frame rate than 30 frames per second increase the size of the video footage only, it does not create any difference in viewing. That is why most of the video recording and transmission are done in 24-30 fps, HDTV uses 50-60 fps for its recording and broadcast, similarly UHDTV uses 100-120 fps. As the resolution of those high speed videos are more than 24-30 fps videos, hence they require faster refresh rate of 50-60 fps and 100-120 fps. Video is recorded using various types of sensors, such as Infrared (passive and active sensors), Optics (video and camera systems), Radio Frequency Energy (radar, microwave and tomography motion detection), Sound (microphones and acoustic sensors), X-Rays sensors, Gamma ray sensors, Ultraviolet sensors. Only camera based sensors gives direct video output which can be seen by our eyes. Other sensors give data which has to be processed and then converted into image format for viewing.

Object tracking is done to know more about the object which is of utmost importance, which we called foreground and all other things are background. Here the main objective is to separate the foreground or object from the background. To

do this first we require a background which is free from foreground objects that is we have to model a background first. Throughout the last couple of decades, several techniques have been introduced to accomplish this task effectively. However there is no perfect system or method which can overcome the various problems that are faced in various situations. The difficulties are generally associated with lighting conditions of the surroundings, illumination of the object itself, shadows of the objects, speed of object movement, shape of the object, type of object, reflection (glass, metal, water), shining of objects in background, waves in water, wakes created by speedboats or ship, effect of wind or breeze on other object present in background (fluttering of flags, flying dry leaves, wriggling of trees branches, movement of leaves etc.). No one technique has been developed to tackle all these challenges; different techniques have been developed for different purposes.

A lot of techniques have been developed towards the Background Modeling. The most simple of them are to take a picture of the background which is free from any foreground activity. But in real life this is not always possible. There are various other techniques to do this. Such as Mean based, Median based, Approximate Median, Kalman Filter, Mixture of Gaussian, Running Gaussian Average method, Non-parametric method, Gaussian Mixture Model etc. yet other types of background extraction algorithms typically use techniques like image in-painting [1] on a single image, texture synthesis [2] to remove foreground objects from still images, a background recovery [3] from a set of images that share an identical background, segmentation-based approaches [4]. However, the method presented in [5] is restricted to rigid moving objects, and the method of [6] relies on differential texture regions to refine the segmentation [7].

After modeling proper background, this background is subtracted from the frames in the video sequence to detect the foreground object. Simultaneously background has to be updated from time to time if has been modeled from the frame

in the video. From time to time situation of background changes due to above discussed challenges, hence it has to be taken into account during the preparation of background, by frequently updating background. The detected or obtained foreground object is continuously tracked by making a selection rectangle around it.

1.2 Importance of Object Detection and Tracking

We humans do the object tracking with our eyes from our existence. Not only us all living thing done it from their existence for one of the most important task of acquiring food. These animals target their prey, track their path and finally hunt them down for food. Even those animals that live on plant products also look for edible item in plant by observing their colour, texture and smell lock and track their target and finally eat them after acquiring it. Even plants and trees track the sunlight for their food preparation. We humans are the biggest user of object detection and tracking in our day to day life. We use it in reading newspaper, reading headlines in television and internet, flying kites, writing, road crossing, walking, cooking, eating etc. it has become a part of our life knowingly or unknowingly.

Now in some place we cannot do tracking by self or wanted automated tracking which we required our machine to do it for us such as aerial surveillance (of vegetation, water bodies, floods, illegal mining activity, enemy position in battle-field, illegal infiltration in border area etc.), study of animal behavior in wild, traffic monitoring in road and in water (river and sea), monitoring terrorist activities (crowded places, important place and buildings), robot vision, computer graphics and animation etc. to name a few.

Object tracking is done to know what is happening with the subject of interest, analysis of the subject is done by tracking its movement. Hence its use in real life application has increased and it is also increasing day by day. We have

discussed few application here such as Aerial Video Surveillance and Monitoring, Traffic Monitoring in Road, Human Monitoring in Busy area (Super Market, Malls, Metro Stations, Roads, Parks, Offices and Housing Apartments etc.), Medical Based Diagnosis (Robotic Surgery, Smart Pill or Radioactive Drug Monitoring), Weather Monitoring (Cloud and Tornado Tracking), Human Computer Interaction (Eyeball or face Tracking, Hand Gestures), Robot Vision and Computer Animation.

A. Aerial video surveillance and monitoring

Surveillance is the monitoring of the behavior, activities, or other changing information, usually of people for the purpose of influencing, managing, directing or protecting them. This can include observation from a distance by means of electronic equipment (such as CCTV cameras), from above the sky (using drones, spy planes or helicopter) or interception (such as Internet traffic or phone calls); and it can include simple, relatively no or low technology methods such as human intelligence agents. Video surveillance data is used by governments for intelligence gathering, the prevention of crime, the protection of public properties, peoples, group or object, or for the investigation of crime. Surveillance is often a violation of privacy, and is opposed by various civil liberties groups and activists.

Aerial surveillance is the gathering of data, usually visual imagery or video from an airborne vehicle such as an unmanned aerial vehicle, helicopter, or spy plane. Military surveillance aircraft use a range of sensors (infrared, thermal imaging, low light camera, electromagnetic sensors) to monitor the battlefield. Government agencies use drones for domestic operations such as security of cities. For instance the MQ-9 Reaper, a U.S. drone plane used for domestic operations by the Department of Homeland Security, carries cameras that are capable of identifying an object the size of milk carton from altitudes of 60,000 feet, and has forward-looking infrared devices that can detect the heat of a human body from a

distance of up to 60 kilometers. Other countries like U.K. is working on a plan to build up a fleet of surveillance UAVs ranging from micro-aerial vehicle to full size drones to be used by the police departments throughout the U.K.

In addition to their surveillance capabilities, MAVs are capable of carrying weapons for crowd control or killing enemy combatants. Programs such as the Heterogeneous Aerial Reconnaissance Team program developed by DARPA have automated much of the aerial surveillance process. They have developed systems consisting of large teams of drone planes that pilot themselves, automatically decide who the 'suspicious' is and how to monitor them, coordinate their activities with other drones nearby, and notify human operators if something suspicious is happening. This greatly increases the amount of area that can be continuously monitored, while reducing the number of human operators required. Researchers are also investigating the possibilities of autonomous surveillance by large groups of micro aerial vehicles stabilized by decentralized bio-inspired swarming rules [8, 9]. In addition to that UAVs are also used to monitor the vegetation and water bodies present in the cities.

B. Traffic monitoring in road

Road traffic is monitored by the government bodies such as local or state authorities. They use CCTV cameras to manage the flow of traffic and provide advice concerning traffic congestion. In automated traffic monitoring system the road signal are controlled by the monitoring system which results in smooth flow of traffic without congestion. These monitoring systems work day and night and in all-weather condition without any problem.

C. Human monitoring in busy area

Human beings are monitored by private and government bodies in various locations. This is done mainly for security reasons for both the humans and property (private and public). After many terrorist attacks on various public places, the monitoring of humans has become important for many government agencies. Now a day those government who are more concerned about their people are installing CCTV cameras all over the cities in important public places, banks, restaurant, hotels, road crossing, parks, hospitals, schools & colleges, airports, docks, railway stations, metro stations, bus stops, offices etc. to curb on miscreants and terrorists. Pedestrian monitoring [10, 11] and crowd control is also done as an important part of human monitoring.

D. Medical based diagnosis

Tracking is use in various streams of medical science such as tomography, smart pill tracking, tracking of eye movement to find mental disorders, robotic surgery. In tomography a radioactive drug is injected inside human organ which is to be diagnosed then that radioactive drug is tracked by computed tomographic (CT) scanner, the scanner receives the radioactive rays and the path of the drug flow is tracked, to get the image of the organ which is to be diagnosed. Several studies have analyzed the link between mental dysfunctions and eye movements, using an eye tracking techniques to determine where a person is looking [12].

E. Weather monitoring

Weather is the state of the atmosphere to the degree that it is hot or cold, wet or dry, calm or stormy, clear or cloudy. Most weather phenomenon occurs in the troposphere just below the stratosphere. It refers to day-to-day temperature and

precipitation activity. It is driven by air pressure, temperature and moisture differences between one place to another. These differences can occur due to the sun's angle at any particular spot, which varies by latitude from the tropics. The strong temperature difference between polar and tropical air gives rise to the jet stream. Due to so many parameters the weather gets constantly changing over time from one place to another. This change of weather for a long period of time gives us climate.

This volatile nature of weather some time creates havoc on us. Hence to reduce catastrophe and calamities monitoring of weather parameters and tracking of low pressure region, wind direction, movement of clouds etc. are required. Which help us to predict the weather condition at a particular place and reduce casualties happened by sunami, typhoons and hurricanes, massive rainfall, excessive snowfall etc., even if the system is not highly accurate it still help us to save our lives.

F. Human computer interaction

Humans interact with computers in many ways; and the interface between humans and the computers they use is crucial to facilitate this interaction. Desktop applications, internet browsers, handheld computers, and computer kiosks make use of the prevalent graphical user interface (GUI) of today. The Association for Computing Machinery (ACM) defines human-computer interaction as "a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them". Due to the multidisciplinary nature of HCI, people with different backgrounds contribute to its success.

People communicate between them using speech hence voice recognition and voice synthesis was the early component in HCI. Gestures, body posture, facial

expressions etc. [13-14] are some of the detail cues used in HCI. All these HCI are slowly being commercialized by the companies such as voice activated operations in mobile devices and in car dashboard, gestures [15] are used in gamming consoles, tracking eye during watching movie, news or browsing internet in TV for giving better user experiences.

G. Robot vision

Robot vision is used for part identification and navigation. Vision applications generally deal with finding a part and orienting it for robotic handling or inspection before an application is performed. Sometimes vision guided robots can replace multiple mechanical tools with a single robot station. Vision ability in robot was due to a combination of vision algorithms, calibration and cameras. Calibration of robot vision is very application dependent. They can range from a simple guidance application to a more complex application that uses data from multiple sensors. Algorithms are constantly improving, allowing for sophisticated detection. Navigation in robot means they have to move along some specific path which was pre-calculated as the most optimal path without being halted by any obstacle or collided with other item present nearby. Path planning [16] is one of the most important tasks in robot navigation. Many robots are now available with collision detection, allowing them to work alongside other robots without the fear of a major collision. They simply stop moving momentarily if they detect another object in their motion path.

1.3 Scope of the thesis

In this thesis, our proposed method of object tracking consists of two parts. Firstly we have prepared a background from the video frame which is free of any moving objects that is foreground. Secondly we have used that background to detect

the foreground object and observe its movement. Finally tracking of moving object has been done by drawing a bounding rectangle around the moving objects of each frame. These two parts are handled separately by two programs written in C.

We have used an Intel based machine with 8 GB RAM and GNU Compiler collection GCC version 5.10; Datasets "EPFL data set: Multi-camera Pedestrian Videos" are collected from website 'http://cvlab.epfl.ch/data/pom'. We have collected 5 different categories of total 18 videos as well as two video from YouTube for testing they are 'www.youtube.com/watch?v=4i_GFrlaStQ' and 'www.youtube.com/watch?v=CfAPnvGFHyM' titled "Sometimes Security Cameras catch a gem!". The experiments show some promising result of the proposed method using the above databases.

1.4 Outline of the thesis

This thesis is organized as follows:

Chapter 2: contains some previous research on background modelling, single and multi-object tracking in video. Numerous techniques have been developed to create proper background from video frames, as well as single and multi-object tracking in video frames.

Chapter 3: presents various approaches to background modelling in video in detail. At first, recursive and non-statistical methods are discussed. Then, non-recursive and statistical methods are given. Thereafter in-image background painting method is discussed for creating background free from any foreground objects.

Chapter 4: presents with various approaches to object tracking in video. Also discuss what various difficulties present for object tracking.

Chapter 5: presents the concept of multi-object tracking in video. How it is different from single object tracking, and what are the challenges in multi-object tracking.

Chapter 6: presents our approach for multi-object tracking in video. Here we discuss how we created the background from one of the frame from video. Then we discuss how we track multiple objects present in the video.

Chapter 7: presents result of the experimentation and discussion. Some images of video frames taken from data sets are shown. Data sets taken from website 'http://cvlab.epfl.ch/data/pom' and YouTube are used.

Chapter 8: presents the conclusion in which we summarized the background modelling and object tracking in video. Future scope of this proposed method has also been mentioned.

Chapter-2

Literature Survey

Now a day object tracking and multi-object tracking has gained a lot of interest from researchers in the field of computer vision. It led to the study of behaviour and analysis of subject of interest to its surroundings in more effective way. The potential of multi-object tracking is increasing day by day in real life due to the availability of low cost hardware, as the data processing requirement is huge. It is commercial used in application such as video surveillance, industrial goods production, security, crowd monitoring, weather forecasting, traffic monitoring, monitoring of animals activity, eye tracking or face tracking in human interface device, medical based diagnosis etc.

Horesh Ben Shitrit, Jérôme Berclaz, François Fleuret and Pascal Fua [17] showed in their paper, that tracking multiple people whose paths may intersect over a long periods of time while retaining their individual identities can be formulated as a convex global optimization problem. Their work is designed to exploit image appearance cues to prevent identity switches. There method is effective even when such cues are only available at distant time intervals. This is unlike with many approaches that depend on appearance being exploitable from frame to frame. As a result, it does better at preserving identity over very long sequences than previous approaches. Furthermore, it depends on a comparatively small number of parameters such as the size of the grid it works on and the maximum number of separate identities to be expected.

Isaac Cohen and Gérard Medioni [18] in their paper address the problem of detection and tracking of moving objects in a video stream obtained from a moving airborne platform. Their proposed method relies on a graph representation of moving objects which allows them to derive and maintain a dynamic template of each moving object by enforcing their temporal coherence. This inferred template along with the graph representation used in their approach allows them to characterize objects trajectories as an optimal path in a graph. Their proposed tracker allows them to deal with partial occlusions, stop and go motion in very challenging situations.

Srenivas Varadarajan, Lina J. Karam, and Dinei Florencio [7] in their paper presents a novel scheme for extracting a still background occluded by a number of foreground objects, moving in different directions and velocities in a video sequence, such that every background pixel is exposed in at least one of the frames. Each identified foreground object is decomposed into blocks. Their scheme is able to efficiently estimate, for each foreground block, a source frame from which the occluded background pixels can be extracted. The pixels of the identified source frames are used to populate the co-located occluded pixels in the initial frame. The efficacy and the simplicity of their algorithm lie in its capacity to recover the background directly from the estimated source frames instead of performing a foreground-background classification for every frame. Their proposed algorithm is robust to variations in lighting and is effective in removing both rigid and deformable foreground objects.

Kuihe Yang, Zhiming Cai, Lingling Zhao [19] showed in their paper that in video surveillance, there are many interference factors such as target changes, complex scenes, and target deformation in the moving object tracking. In order to resolve these issues, they have done the comparative analysis of several common moving object detection methods. They presented a moving object detection and

recognition algorithm that combined frame difference with background subtraction. In their algorithm, they first calculate the average of the values of the gray of the continuous multi-frame image in the dynamic image, and then get the background image by the statistical average of the continuous image sequence, that is, the continuous interception of the N-frame images are summed, to find their average. They also showed that, weight of object information has been increasing, and also restrained the static background. Eventually the motion detection image contains both the target contour and more target information of the target contour point from the background image. In this way they achieve separating the moving target from the image.

Muyun Weng, Guoce Huang and Xinyu Da [20] showed in their paper, a new inter-frame difference algorithm for moving target detection, which is under a static background based on three-frame-difference method in combination with background subtraction method. In their method Firstly, the current frame image subtracts the previous frame and the next frame image separately, their results are added together to get a gray-scale image of the three-frame-difference method. Secondly, the current frame image subtracts the background image to get another gray-scale image of background subtraction method. Thirdly, sum of the two gray-scale images of above is translated into binary image after being judged by threshold. Finally, that binary image is processed by morphology filtering and connectivity analyzing. By this way they obtained the moving region. Their proposed new algorithm takes advantage of the good performances of three-frame-difference method and background subtraction method adequately.

François Brémond and Monique Thonnat [21] showed a method to track multiple non-rigid objects in video sequences. They use the notion of target to represent the perception of object motion. To handle the particularities of non-rigid objects they define a target as an individually tracked moving region or as a group of

moving regions globally tracked. They explained how to compute the trajectory of a target and how to compute the correspondences between known targets and newly detected moving regions. In the case of an ambiguous correspondence they defined a compound target to freeze the associations between targets and moving regions until a more accurate information is available.

Hyungki Roh, Seonghoon Kang and Seong-Whan Lee [22] showed a method for the detection and tracking of multiple people totally occluded or out of sight in a scene for some period of time in image sequences. Their approach is to use time weighted color information, that is the temporal color, for robust multiple people tracking in short- and mid-term tracking. Although the position, shape, and velocity are suitable features in tracking in consecutive frames, they cannot track the target continuously when the target disappears temporarily. Since their proposed temporal color is accumulated with its associated weight, the target can be continuously tracked even when the target is occluded or leaves the scene for a few seconds or minutes. It assures that the system will continuously track people moving in a group with occlusion. They showed that the temporal color is more stable than shape or intensity when used in various cases. Problems with temporal color only occur when people are in uniforms or clothes with similar color.

Yoshinori Ohno, Jun Miura, and Yoshiaki Shirai [23] in soccer games, understanding the movement of players and a ball is essential for the analysis of matches or tactics. In their paper, they discussed a system to track players and a ball and to estimate their positions from video images. Their system tracks players by extracting shirt and pants regions and can cope with the posture change and occlusion by considering their colors, positions, and velocities in the image. Their system extracts ball and candidates by using the color and motion information, and determines the ball among them based on motion continuity. To determine the player who was holding the ball, the position of players on the field and the 3D

position of the ball are estimated. They showed that the ball position is estimated by fitting a physical model of movement in the 3D space to the observed ball trajectory.

Sohaib Khan, Omar Javed, Zeeshan Rasheed, Mubarak Shah [24] multiple cameras are needed to cover large environments for monitoring activity, to track people successfully in multiple perspective imagery, one needs to establish correspondence between objects captured in multiple cameras. They presented a system for tracking people in multiple uncalibrated cameras. The system is able to discover spatial relationships between the cameras fields of view and use this information to correspond between different perspective views of the same person. They employed a novel approach of finding the limits of field of view (FOV) of a camera as visible in the other cameras. Using this information, when a person is seen in one camera, they are able to predict all the other cameras in which this person will be visible. They have described a framework to solve the camera handoff problem. They contend that the camera calibration and 3D reconstruction is unnecessary for solving this problem. Instead, they presented with a system based on edge of FOV lines of cameras that can handle handoffs. They outline a process to automatically find the lines representing these limits, and then using them to resolve the ambiguity between multiple tracks. Their approach does not require feature matching, which is difficult in widely separated cameras.

Koichi Sato and J. K. Aggarwal [25] in their paper present a methodology for tracking persons and identifying two-person interactions in outdoor image sequences. By locating and tracking two persons over consecutive frames of monocular grayscale image sequences, they classify their interactions into several classes. Some of the interaction classes are: One person leaves another stationary person; two persons meet coming from different directions; and one stationary person starts following another walking person. They used side-view image sequences obtained by a fixed camera. In these image sequences the subjects are

frequently occluded and move perpendicular to the direction of the camera. Their presented system can accurately recognize 9 different interactions automatically. Due to the human extraction and the temporal spatio-velocity transform, their system performs robustly even on low quality images such as outdoor images, dark images, images that contain small human blobs and images that contain humans with similar intensity to the background intensity. Their proposed system has two limitations. (1) It cannot distinguish two people who are positioned too close to each other, because of too much occlusion. In that case, the interaction is recognized as one person's activity. (2) The system assumes that the interactions should occur in a sidewalk type of situation. It cannot recognize movement along the camera direction. If it finds such movement, it is considers to be stationary.

Shiloh L. Dockstadert and A. Murat Tekalp [26] proposed a distributed, real-time computing platform for tracking multiple interacting persons in motion. To overcome occlusion and articulated motion they used a multi-view implementation, where 2D semantic features are independently tracked in each view and then collectively integrated using a Bayesian belief network with a topology that varies as a function of scene content and feature confidence. Their network fuses observations from multiple cameras by resolving independency relationships and confidence levels within the graph, thereby producing the most likely vector of 3D state estimates given the available data. They demonstrated the efficacy of the proposed system using a multi-view sequence of several people in motion. They suggest that, when compared with data fusion based on averaging, their proposed technique yields a noticeable improvement in tracking accuracy.

Dieter Koller, Joseph Weber and Jitendra Malik [27] they propose a new approach for detecting and tracking vehicles in road traffic scenes that attains a level of accuracy and reliability. To get high accuracy and reliability they employed a contour tracker based on intensity and motion boundaries. The motion of the contour

of the vehicles in the image is assumed to be well describable by an affine motion model with a translation and a change in scale. A contour associated to a moving region is initialized using a motion segmentation step which is based on image differencing between an acquired image and a continuously updated background image. A vehicle contour is represented by a closed cubic spline the position and motion of which is estimated along the image sequence. In order to employ linear Kalman Filters they decompose the estimation process in two filters: one for estimating the affine motion parameters and one for estimating the shape of the contours of the vehicles. Occlusion detection is performed by intersecting the depth ordered regions associated to the objects. The intersection is then excluded in the motion and shape estimation. Their procedure also improves the shape estimation in case of adjacent objects since occlusion detection is performed on slightly enlarged regions. In this way they obtain robust motion estimates and trajectories for vehicles even in the case of occlusions.

Chapter-3

Background Modeling

Background modeling also called background preparation, is a method by which we create a background free of any foreground objects. This background is further use to detect foreground object in video by using the method of background subtraction. Background subtraction, also known as Foreground Detection, is a technique in the fields of image processing and computer vision wherein an image's foreground is extracted for further processing (object recognition, tracking etc.). Generally an image's regions of interest are objects such as humans, vehicles, text, animals, bacteria, stars and galaxies etc. in its foreground. Background subtraction is a widely used approach for detecting moving objects in videos from static cameras. The rationale in the approach is that of detecting the moving objects from the difference between the current frame and a reference frame, often called “background image”, or “background model”.

Here backgrounds are of two types static and dynamic. Static backgrounds are backgrounds where all objects which consist of background are still i.e. not moving. Whereas in dynamic backgrounds not all objects which are termed as background are still for example wriggling of branches and leaves of trees, fluttering of flags or banner, waves in water, clouds movement, flying objects (dry leaves, paper, dust, flying insects etc.). Static backgrounds are the most preferred background in background modeling whereas dynamic backgrounds are difficult to model. However, getting a static background free of any foreground objects or activities is generally difficult to get in real environments. The real challenge lies in tackling the whether effect, level of brightness, day and night view, wind effect,

movement of camera, shaking effect, reflection, occlusion and so on. Different approaches are invented to tackle these effects; but no one is full proof, it is still a major research area to create an efficient algorithm to tackle above problems.

3.1 Types of Background modeling

Background modeling is the heart of any background subtraction algorithm. Much research has been devoted to developing a background model that is robust against environmental changes in the background, but sensitive enough to identify all moving objects of interest. Background modeling techniques are classified into two categories Non-recursive method and Recursive method and can also be classified into two other categories such as Non-statistical method and Statistical method based on the complexity of the algorithms used.

I. Non-recursive method: This technique uses a sliding-window approach for background estimation. It stores a buffer of the previous video frames and estimates the background image based on the temporal variation of each pixel within the buffer. Non-recursive techniques are highly adaptive as they do not depend on the history beyond those frames stored in the buffer. Static Frame based method, Mean Value method, Median Value method, Linear Predictive method and Non Parametric model are some examples of non-recursive algorithms.

II. Recursive method: These techniques do not maintain a buffer for background estimation. Instead, they recursively update a single background model based on each input frame. As a result, input frames from distant past could have an effect on the current background model. Compared with non-recursive techniques, recursive techniques require less storage, but any error in the background model can stay for a much longer period of time. Some examples of algorithms found in this category

are: Approximated Median Value method, Kalman Filter method and Mixture of Gaussians method.

A. Non-statistical method: In this approach of background modeling, background is created by information of pixels from current frame and its past frames in the video sequences. The non-statistical methods are suitable for real time applications as they are considerably fast, but it is not helpful in modeling dynamic backgrounds. Various non-statistical based background modeling techniques are Static Frame based method, Mean Value method, Median Value method, Approximated Median Filter, Kalman Filter, Running Gaussian Average method etc.

B. Statistical method: In statistical-based background modeling, the probability function of background is estimated. This function determines the probability for the belonging of the pixel to the background or foreground. Despite non-statistical based methods, these approaches are suitable for modeling outdoor and dynamic scenes. Various statistical based background modeling techniques are Gaussian or Background Mixture Model, Non-Parametric Model, Mixture of Gaussian method etc.

3.2 Different methods use in background modeling

3.2.1 Static Frame Based Method

This is the most simplest of all method, here a frame free of any foreground objects is taken as background. This frame is use for detecting the foreground objects from video by frame differencing method. This background is use for entire length of the video. The background is basically the first frame of the video or taken as a separate frame image which does not have any foreground objects.

This simplicity brings a lot of problem, as same background cannot be used in other situations such as day and night, morning and evening, in rain, in foggy or cloudy weather etc. to use it in different situation background image frame from different situation has to be kept aside from beginning, which is a cumbersome process. It is not possible to store images from each and every situation. Hence this method is not used in real life situations it is only used in laboratory condition for experimentation.

3.2.2 Mean Value Method

In the mean value method for calculating the background image, a series of preceding images are averaged from current frame. In this method the background image get constantly updated after N number of frames. Here any changes in the background get constantly updated in the background image. Calculating the background image at the instant t, where N is the number of preceding images taken for averaging.

$$B(x,y) = \frac{1}{N}\sum_{i=1}^{N} V(x,y,t-i) \qquad 3.1$$

This averaging refers to averaging corresponding pixels in the given images. N would depend on the video speed (number of images per second in the video) and the amount of movement in the video.

This method is better than taking single image as background, as background free of foreground activity is not always easy to achieve. Also background gets constantly updated after N frames hence no need to keep separate images for different situations. The disadvantage of this method is that it will create distortion in the background model if large change in the pixel intensity present in the frames.

3.2.3 Median Value Method

In the median value method we calculate the median of N previous frames from the current frame for modeling the background. Here also background gets updated after N number of frames. This method is similar to the **mean value method** but we take median of pixels instead of mean. Let P represent a video sequence having N image frames. The background B(x, y) can be constructed using the formula:

$$B(x,y) = median[P1(x,y), P2(x,y), \dots PN(x,y)] \quad 3.2$$

The value of B(x, y) is the background brightness to be calculated in the pixel location (x, y) and median symbolizes its median value of pixel intensities over N frames.

This method is better than previous two methods because it do not require any static background to be kept for different situations. And it is better than mean value method because it is immune to spurious values when there is large change in pixel intensities, and it is faster to compute than creating mean of pixel intensities.

3.2.4 Approximate Median Filter

Due to the success of non-recursive median filtering, McFarlane and Schofield propose a simple recursive filter to estimate the median. This technique has also been used in background modeling for urban traffic monitoring. In this scheme, the running estimate of the median is incremented by one if the input pixel is larger than the estimate, and decreased by one if smaller than the estimate.

$$if\ (Fr(x,y)_{i-1} > Bg(x,y)_{i-1}) \rightarrow Bg(x,y)_i = Bg(x,y)_{i-1} + 1; \quad 3.3$$

$$if\ (Fr(x,y)_{i-1} < Bg(x,y)_{i-1}) \rightarrow Bg(x,y)_i = Bg(x,y)_{i-1} - 1; \quad 3.4$$

This estimate eventually converges to a value for which half of the input pixels are larger than and half are smaller than this value, that is, the median.

The main problem with the Approximate Median Filter is its slow recovery to changes in the background. It also has a predetermined threshold and ignores any correlation between neighboring pixels.

3.2.5 Kalman Filter

Kalman filter, also known as linear quadratic estimation (LQE), is an algorithm that uses a series of measurements observed over time, containing statistical noise and other inaccuracies, and produces estimates of unknown variables that tend to be more precise than those based on a single measurement alone, by using Bayesian inference and estimating a joint probability distribution over the variables for each timeframe. This filter is named after Rudolf E. Kálmán, one of the primary developers of its theory.

The Kalman filter has numerous applications in technology. A common application is for guidance, navigation and control of vehicles, particularly aircraft and spacecraft. Furthermore, the Kalman filter is a widely applied concept in time series analysis used in fields such as signal processing and econometrics. Kalman filters also are one of the main topics in the field of robotic motion planning and control, and they are sometimes included in trajectory optimization. The Kalman filter has also found use in modeling the central nervous system's control of movement. Due to the time delay between issuing motor commands and receiving sensory feedback, use of the Kalman filter provides the needed model for making estimates of the current state of the motor system and issuing updated commands.

The algorithm works in a two-step process. In the prediction step, the Kalman filter produces estimates of the current state variables, along with their uncertainties. Once the outcome of the next measurement (necessarily corrupted with some amount of error, including random noise) is observed, these estimates are updated using a weighted average, with more weight being given to estimates with higher certainty. The algorithm is recursive. It can run in real time, using only the present input measurements and the previously calculated state and its uncertainty matrix; no additional past information is required.

It is a recursive technique for tracking linear dynamical systems under Gaussian noise. Many different versions have been proposed for background modeling, differing mainly in the state spaces used for tracking. The simplest version uses only the luminance intensity. Karmann and von Brandt use both the intensity and its temporal derivative, while Koller, Weber, and Malik use the intensity and its spatial derivatives. We provide a brief description of the popular scheme used in. The internal state of the system is described by the background intensity B_t and its temporal derivative B'_t, which are recursively updated as follows:

$$\begin{bmatrix} B_t \\ B'_t \end{bmatrix} = \boldsymbol{A} \cdot \begin{bmatrix} B_{t-1} \\ B'_{t-1} \end{bmatrix} + \boldsymbol{K_t} \cdot \left(I_t - \boldsymbol{H} \cdot \boldsymbol{A} \cdot \begin{bmatrix} B_{t-1} \\ B'_{t-1} \end{bmatrix} \right) \qquad 3.5$$

Matrix A describes the background dynamics and H is the measurement matrix. Their particular values used in are as follows:

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0.7 \\ 0 & 0.7 \end{bmatrix}, \qquad \boldsymbol{H} = \begin{bmatrix} 1 & 0 \end{bmatrix} \qquad 3.6$$

The Kalman gain matrix K_t switches between a slow adaptation rate α1 and a fast adaptation rate α2 > α1 based on whether I_{t-1} is a foreground pixel:

$$K_t = \begin{bmatrix}\alpha_1\\ \alpha_1\end{bmatrix} \; if \; I_{t-1} \; is \; foreground, and \; \begin{bmatrix}\alpha_2\\ \alpha_2\end{bmatrix} \; otherwise. \qquad 3.7$$

3.2.6 Running Gaussian Average Method

For this method, Wren et al. propose fitting a Gaussian probabilistic density function (pdf) on the most recent n frames. In order to avoid fitting the pdf from scratch at each new frame time *t*, a running (or on-line cumulative) average is computed.

The pdf of every pixel is characterized by mean μ_t and variance σ_t^2. The following is a possible initial condition (assuming that initially every pixel is background):

$$\mu_0 = I_0 \qquad 3.8$$

$$\sigma_0^2 = < some \; default \; value > \qquad 3.9$$

Where I_t is the value of the pixel's intensity at time *t*. In order to initialize variance, we can, for example, use the variance in *x* and *y* from a small window around each pixel.

Note that background may change over time (e.g. due to illumination changes or non-static background objects). To accommodate for that change, at every frame *t*, every pixel's mean and variance must be updated, as follows:

$$\mu_t = \rho I_t + (1 - \rho)\mu_{t-1} \qquad 3.10$$

$$\sigma_t^2 = d^2\rho + (1 - \rho)\sigma_{t-1}^2 \qquad 3.11$$

$$d = |(I_t - \mu_t)| \qquad 3.12$$

Where ρ determines the size of the temporal window that is used to fit the pdf (usually ρ = 0.01) and d is the Euclidean distance between the mean and the value of the pixel.

We can now classify a pixel as background if its current intensity lies within some confidence interval of its distribution's mean:

$$\frac{|(I_t - \mu_t)|}{\sigma_t} > k \longrightarrow Foreground \tag{3.13}$$

$$\frac{|(I_t - \mu_t)|}{\sigma_t} \leq k \longrightarrow Background \tag{3.14}$$

Where the parameter k is a free threshold (usually $= 2.5$). A larger value for k allows for more dynamic background, while a smaller k increases the probability of a transition from background to foreground due to more subtle changes.

In a variant of the method, a pixel's distribution is only updated if it is classified as background. This is to prevent newly introduced foreground objects from fading into the background. The update formula for the mean is changed accordingly:

$$\mu_t = M\mu_{t-1} + (1 - M)(I_t\rho + (1 - \rho)\mu_{t-1}) \tag{3.15}$$

Where $M = 1$ when I_t is considered foreground and $M = 0$ otherwise. So when $M = 1$, that is, when the pixel is detected as foreground, the mean will stay the same. As a result, a pixel, once it has become foreground, can only become background again when the intensity value gets close to what it was before turning foreground. This method, however, has several issues: It only works if all pixels are initially background pixels (or foreground pixels are annotated as such). Also, it cannot cope with gradual background changes: If a pixel is categorized as foreground for a too long period of

time, the background intensity in that location might have changed (because illumination has changed etc.). As a result, once the foreground object is gone, the new background intensity might not be recognized as such anymore.

3.2.7 Gaussian or Background Mixture Model

One of the challenging issues in background modeling is to model repetitive motions in the video such as the shining water, leaves of a branch, or a waving flag. Stauffer and Grimson in have introduced the Gaussian mixture model (GMM) to extract the statistics of repetitive moving objects often exist in outdoor scenes.

In this technique, it is assumed that every pixel's intensity values in the video can be modeled using a Gaussian mixture model. A simple heuristic determines which intensities are most probably of the background. Then the pixels which do not match to these are called the foreground pixels. Foreground pixels are grouped using 2D connected component analysis. At any time t, a particular pixel (x_0, y_0)'s history is

$$X_1, \dots, X_t = \{V(x_0, y_0, i) : 1 \leq i \leq t\} \quad 3.16$$

This history is modeled by a mixture of K Gaussian distributions:

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,t} N\left(X_t \mid \mu_{i,t}, \Sigma_{i,t}\right) \quad 3.17$$

Where $\omega_{i,t}$ denotes the weight for the i-th kernel, and $\Sigma_{i,t}$ as the covariance parameter and

$$N\left(X_t \mid \mu_{i,t}, \Sigma_{i,t}\right) = \frac{1}{(2\pi)^{D/2}} \frac{1}{\left|\Sigma_{i,t}\right|^{1/2}} \exp\left(-\frac{1}{2}\left(X_t - \mu_{i,t}\right)^T \Sigma_{i,t}^{-1}\left(X_t - \mu_{i,t}\right)\right) \quad 3.18$$

The function $N\left(X_t \mid \mu_{i,t}, \Sigma_{i,t}\right)$ is the normal density function for feature vector X_t of i-th kernel. The update parameters are calculated as given below:

$$w_{k+1} = (1-\alpha)w_k + \alpha \cdot P(k \mid x_t, \mu_k, \Sigma_k) \quad 3.19$$

$$\mu_{k+1} = (1 - \rho_k)\mu_k + \rho_k x_t \quad 3.20$$

$$\Sigma_{k+1} = (1 - \rho_k)\Sigma_k + \rho_k(x_t - \mu_{k+1})(x_t - \mu_{k+1})^T \quad 3.21$$

$$\alpha = \frac{1}{N+1} \quad 3.22$$

$$\rho_k = \frac{\alpha \cdot P(k \mid x_t, \mu_k, \Sigma_k)}{w_k} \quad 3.23$$

Where w is a weight factor called the mixing coefficients, $P(k \mid x_t, \mu_k, \Sigma_k)$ is a normal function, μ is mean, Σ is covariance, N is number of frames, k is the k-th frame, ρ is the learning rate for the parameters, α is a user-defined learning rate with values $0 \leq \alpha \leq 1$ and x_t is the feature vector or current frame. Or an on-line K-means approximation is used to update the Gaussians. Numerous improvements of this original method developed by Stauffer and Grimson have been proposed and a complete survey can be found in Bouwmans et al.

3.2.8 Non-Parametric Model

In this approach, there is no need to optimize the parameters of each kernel. For modeling the messy and fast wiggling behavior, the model must be updated continuously in order to capture the fast changes in the scene background.

For describing this model, let $x_1, x_2, \dots, x_N$ be a recent sample of intensity values for a pixel. The probability density function, which indicates the pixel intensity value (x_t) at time t, can be estimated using kernel estimator K as following:

$$\Pr(x_t) = \frac{1}{N}\sum_{i=1}^{N} K(x_t - x_i) \qquad 3.24$$

If we choose our kernel estimator function, K to be a Normal function N(0, Σ), where Σ represents the bandwidth of kernel function, then the density can be estimated using below equation:

$$\Pr(x_t) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{(2\pi)^{\frac{d}{2}}\sqrt{|\Sigma|}}\, e^{-\frac{1}{2}(x_t - x_i)^T \Sigma^{-1}(x_t - x_i)} \qquad 3.25$$

If we assume independency between the different colour channels, and each colour channel (j-th channel) has a different kernel bandwidth value of σ_i^2, then the bandwidth matrix would be:

$$\Sigma = \begin{pmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \sigma_2^2 & 0 \\ 0 & 0 & \sigma_3^2 \end{pmatrix} \qquad 3.26$$

And the density estimation is reduced to:

$$\Pr(x_t) = \frac{1}{N}\sum_{i=1}^{N}\prod_{j=1}^{d} \frac{1}{\sqrt{2\pi\sigma_j^2}}\, e^{-\frac{(x_{tj} - x_{ij})^2}{2\sigma_j^2}} \qquad 3.27$$

The pixel x_t is considered as part of foreground pixel if $\Pr(x_t) > t$ where the threshold t is a global threshold over the entire image that can be adjusted to achieve a desired accuracy.

Since we measure the deviations between two consecutive intensity values, the pair (x_i, x_j) usually comes from the same local-in-time distribution and only few pairs are expected to come from cross distributions. If we assume that this local-in-time distribution is Normal (μ, σ^2), then the deviation (x_i - x_j) has also a Normal distribution with N(μ, $2\sigma^2$). Therefore, the standard deviation of the first distribution can be estimated as:

$$\sigma = \frac{1}{0.68\sqrt{m}} \qquad 3.28$$

3.2.9 Mixture of Gaussian

First, each pixel is characterized by its intensity in RGB color space. Then probability of observing the current pixel is given by the following formula in the multidimensional case

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,t} \eta \left(X_t \, \mu_{i,t}, \Sigma_{i,t}\right) \qquad 3.29$$

Where the parameters are K is the number of distributions, ω is a weight associated to the i^{th} Gaussian at time t with mean μ and standard deviation Σ.

$$\eta\left(X_t \, \mu_{i,t}, \Sigma_{i,t}\right) = \frac{1}{(2/\pi)^{n/2} \Sigma_{i,t}^{0.5}} \, exp\left(-\frac{1}{2}\left(X_t - \mu_{i,t}\right)\Sigma_{i,t}\left(X_t - \mu_{i,t}\right)\right) \qquad 3.30$$

Once the parameters initialization is made, a first foreground detection can be made then the parameters are updated. The first B Gaussian distribution which exceeds the threshold T is retained for a background distribution

$$B = argmin\left(\Sigma_{i=1}^{B}\ \omega_{i,t}\ > T\right) \qquad 3.31$$

The other distributions are considered to represent a foreground distribution. Then, when the new frame comes at times $t + 1$, a match test is made of each pixel. A pixel matches a Gaussian distribution if the Mahalanobis distance.

$$\left((X_{t+1} - \mu_{t+1})^T\ \Sigma_{i-1}^{b}\ (X_{t+1} - \mu_{t+1})\right)^{0.5}\ < k * \sigma_{i,t} \qquad 3.32$$

Where k is a constant threshold which equal to 2.5. Then, two cases can occur:

Case 1: A match is found with one of the K Gaussians. For the matched component, the update is done as follows

$$\sigma_{i,t+1} = (1 - \rho)\ \sigma_{i,t}^2 + \rho(X_{x+1} - \mu_{x+1})(X_{x+1} - \mu_{x+1})^T \qquad 3.33$$

Power and Schoonees used the same algorithm to segment the foreground of the image

$$\sigma_{i,t+1} = (1 - \alpha)\omega_{i,t} + \alpha P(k \mid X_t, \phi) \qquad 3.34$$

The essential approximation to $P(k \mid X_t, \phi)$ is given by $M(k, t)$

$$M(k,t) = 1\ (match), M(k,t) = 0\ (otherwise) \qquad 3.35$$

Case 2: No match is found with any of the K Gaussians. In this case, the least probable distribution K is replaced with a new one with parameters

$$k_{i,t} = lowPriorWeight \tag{3.36}$$

$$\mu_{i,t+1} = X_{t+1} \tag{3.37}$$

$$k_{i,t+1} = LargeInitialWeight \tag{3.38}$$

Once the parameter is maintained, foreground detection can be made and so on.

3.2.10 Background synthesis

Several approaches have been proposed in the past for locating the foreground regions in video sequences [5, 31]. Though these techniques perform well in the basic foreground-background classification on a per-image basis, they do not recover the occluded background. Inpainting techniques [28] have been used to fill in the occluded areas by suitably interpolating the neighboring pixels or using texture synthesis [32] to remove foreground objects from still images. These techniques can result in a degraded visual quality when applied to video sequences because, instead of extracting the real value of the exposed pixel, they try to form an estimate.

C. Herley [33] in his paper proposed a method of background recovery from a set of images that share an identical background. His method can only be applied to a video sequence on a frame-by-frame basis as it does not exploit the temporal correlation that is present in the video sequence. When applied to a video sequence, his method would consider every frame as a potential source for un-occluding every occluded region and would lead to excessive computations.

Segmentation-based approaches have also been proposed [34, 35]. However, the method presented in [34] is restricted to rigid moving objects, and the method of [35] relies on differential texture regions to refine the segmentation. A set of motion-based approaches [29, 30, 36, 37] have also been proposed, many of which do simultaneously foreground tracking and background updating. But these methods are computationally very expensive as the foreground-background classification is done for every frame

Srenivas Varadarajan et al [7] in their paper present a video-based background extraction scheme. The proposed scheme estimates a source frame for every foreground block, from which the occluded background pixels can be extracted and is suitable for real-time background recovery.

Chapter-4

Object Tracking in a Video

Object Detection and tracking is considered as important subject within the area of computer vision. Availability of high definition videos, fast processing computers and exponentially increasing demand for highly reliable automated video analysis has created a new and great deal for modifying object tracking algorithms. Video analysis has three main steps mainly: interesting moving objects detection, the tracking the object detected from frame to frame and visualizing and analyzing to identify the behavior of the object in the entire video.

4.1 Steps of Object Tracking

Object tracking is simply performed in following two steps:
1. Object detection.
2. Object tracking.

4.1.1 Object Detection

Object detection is the process in which we choose the subject of our interest present in a video sequence. Object detection is done by subtracting the background from the current video frame, which is done by the process of background modeling. Background modeling is the basis of any foreground detecting algorithm. Most of the research work is devoted toward proper background modeling which can handle the new challenges.

4.1.2 Object Tracking

Tracking is the problem of estimating the trajectory of an object as it moves around a scene. Ultimate aim of object tracking is to associate objects targeted in upcoming video frames. This can be very difficult in case relative motion between moving objects and frame rate is high.

Change of orientation of tracked object with passing of time can increase the complexity of the process. Thus to deal with the above problem we are employ a model for motion of the object to show how the image of the target might get change for every possible motion of the object.

Basically tracker assigns consistent and unique labels to the objects tracked in different frames of a video. Additionally, depending on the domain of tracking, a tracker can also provide centroid information of the object, orientation, shape, or are of an object.

4.2 Algorithms used in object tracking

To perform object tracking in video an algorithm analyzes sequential video frames and outputs the movement of targets between the frames. There are a variety of algorithms, each having strengths and weaknesses. Considering the intended use is important when choosing an algorithm to use. There are two major components of a visual tracking system: target representation and localization, as well as filtering and data association.

Target representation and localization is mostly a bottom-up process. These methods give a variety of tools for identifying the moving object. Locating and tracking the target object successfully is dependent on the algorithm. For example,

using blob tracking is useful for identifying human movement because a person's profile changes dynamically [38] with respect to time. Typically the computational complexity for these algorithms is low. The following are some common target representation and localization algorithms:

- **Kernel-based tracking (mean-shift tracking [39]):** An iterative localization procedure based on the maximization of a similarity measure (Bhattacharyya coefficient).

- **Contour tracking:** Detection of object boundary (e.g. active contours or Condensation algorithm). Contour tracking methods iteratively evolve an initial contour initialized from the previous frame to its new position in the current frame. This approach to contour tracking directly evolves the contour by minimizing the contour energy using gradient descent.

Filtering and data association is mostly a top-down process, which involves incorporating prior information about the scene or object, dealing with object dynamics, and evaluation of different hypotheses. These methods allow the tracking of complex objects along with more complex object interaction like tracking objects moving behind obstructions [40]. Additionally the complexity is increased if the video tracker (also named TV tracker or target tracker) is not mounted on rigid foundation (on-shore) but on a moving ship (off-shore), where typically an inertial measurement system is used to pre-stabilize the video tracker to reduce the required dynamics and bandwidth of the camera system. The computational complexity for these algorithms is usually much higher. The following are some common filtering algorithms:

- **Kalman filter:** It is an optimal recursive Bayesian filter for linear functions subjected to Gaussian noise. It is an algorithm that uses a series of measurements observed over time, containing noise (random variations) and other inaccuracies,

and produces estimates of unknown variables that tend to be more precise than those based on a single measurement alone [41].

• **Particle filter:** useful for sampling the underlying state-space distribution of nonlinear and non-Gaussian processes [42].

4.3 Difficulties in Object Tracking

Tracking objects can become complex and difficult due to following reasons:

- Projection of the 3D world on a 2D image will cause loss of information.
- Presence of noises in the image.
- Irregular object motion.
- Non rigid nature of objects.
- Occurrence of occlusion (partial or full).
- Objects shape is complex.
- Change in the illumination or intensity of the scene.

Chapter-5

Multi-Object Tracking in a Video

Multiple Object Tracking (MOT) is an important computer vision task which has gained increasing attention due to its academic and commercial potential. Although different kinds of approaches have been proposed to tackle this problem, there still exist many issues unsolved. For example, factors such as abrupt appearance changes and severe object occlusions pose great challenges for MOT. The task of MOT is largely partitioned to locating multiple objects, maintaining their identities and yielding their individual trajectories given an input video.

Objects to track can be, for example, pedestrians on the street [43, 44], vehicles [45, 46], sports players in the court [47, 48, 49], or a flock of animals, (birds [50], bats [51], ants [52], fishes [53, 54], cells [55, 56], etc.). The multiple 'objects' could also be different parts of a single object. As a mid-level task in computer vision, multiple object tracking grounds high-level tasks such as action recognition, behavior analysis, etc. It has numerous applications. Some of them are presented in the following.

A. Visual Surveillance: The massive amount of videos, especially surveillance videos, requires automatic analysis to detect abnormal behaviors, which is based on analyses of objects actions, trajectories, etc. To obtain such information, we need to locate targets and track them, which is exactly the objective of multiple object tracking.

B. Human Computer Interface (HCI): Visual information, such as expression, gesture, can be employed to achieve advanced HCI. Extraction of visual information requires visual tracking as the basis. When multiple objects appear in the scene, we need to consider interactions among them. In this case, MOT plays a crucial rule to make the HCI more natural and intelligent.

C. Virtual Augment Reality (VAR): MOT also has an application for this problem. For instance, MOT can supply users with better experience in video conferences.

D. Medical Image Processing: Some tasks of medical image processing require laborious manual labeling. For instance, labeling multiple cells in images. In this case, MOT can help to save a large amount of labeling cost.

The various applications above have sparkled enormous interest in this topic. However, compared with Single Object Tracking (SOT) which primarily focuses on designing sophisticated appearance models or motion models to deal with challenging factors such as scale changes, out-of-plane rotation and illumination variations, multiple object tracking additionally requires maintaining the identities among multiple objects. Besides the common challenges in both SOT and MOT, the further key issues making MOT challenging include (but not limit to):

i. Frequent occlusions.
ii. Initialization and termination of tracks.
iii. Small size of objects [51].
iv. Similar appearance among objects.
v. Interaction among multiple objects.

5.1 Overview of multi-object tracking

In order to deal with the MOT problem, a wide range of solutions have been proposed in recent years. These solutions focus on different aspects of a MOT system, making it difficult for researchers. The objective of MOT is to produce trajectories of objects as they move around the image plane. Multiple object tracking can generally be formulated as a multi-variable estimation problem. Given an image sequence $\{I_1, I_2, \ldots, I_t, \ldots\}$ as input, we employ s_t^i to denote the state of the *i*-th object in the *t*-th frame. We use $S_t = (s_t^1, s_t^2, \ldots, s_t^{M_t})$ to denote states of all the M_t objects in the *t*-th frame, $s_{1:t}^i = \{s_1^i, s_2^i, \ldots, s_t^i\}$ to denote the sequential states of the *i*-th object from the first frame to the *t*-th frame, and $s_{1:t} = \{s_1, s_2, \ldots, s_t\}$ to denote all the sequential states of all the objects from the first frame to the t-th frame. Note that the object number may vary from frame to frame. To estimate the states of objects, we need to collect some observations from the image sequence. Correspondingly, we utilize o_t^i to denote the collected observations for the *i*-th object in the *t*-th frame, $O_t = (o_t^1, o_t^2, \ldots, o_t^{M_t})$ to denote the collected observations for all the M_t objects in the *t*-th frame, $o_{1:t}^i = \{o_1^i, o_2^i, \ldots, o_t^i\}$ to denote the sequential observations collected from the first frame to the *t*-th frame, and $O_{1:t} = \{O_1, O_2, \ldots, O_t\}$ to denote all the collected sequential observations of all the objects from the first frame to the t-th frame. The objective of multiple object tracking is to find the 'optimal' sequential states of all the objects, which can be generally modeled by performing MAP (maximal a posterior) estimation from the conditional distribution of the sequential states of all the objects given all the observations:

$$\hat{S}_{1:t} = arg_{S_{1:t}} \max P(S_{1:t}|O_{1:t}) \qquad 5.1$$

The estimation can be performed using probabilistic inference algorithms based on a two-step iterative procedure [57, 58, 59, 60, 61, 62, 63]

$$Predict: P(S_t|O_{1:t-1}) = \int P(S_t|S_{t-1})\, P(S_{t-1}|O_{1:t-1})\, dS_{t-1} \quad 5.2$$

$$Update: P(S_t|O_{1:t}) \propto P(O_t|S_t)\, P(S_t|O_{1:t-1}) \quad 5.3$$

In the formula above, $P(S_t|S_{t-1})$ and $P(O_t|S_t)$ are the Dynamic Model and the Observation Model, respectively. These two models play a very important role in a tracking algorithm. Since the distributions of these two models are usually unknown, sampling methods like Particle Filter [64, 65, 66, 67, 68, 57, 58, 59], MCMC [52, 69, 70], RJMCMC [71] etc. are employed to perform the estimation.

5.2 Function of Multi-object tracking

5.2.1 Initialization Method

The first criterion is that how objects are initialized. According to this criterion, most of existing MOT work could be grouped into two sets [72] Detection Based Tracking (DBT) and Detection Free Tracking (DFT). DBT relies on object detection while DFT does not.

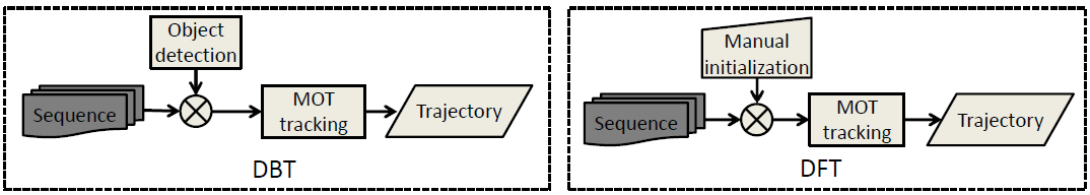

Fig. 1: Procedure flow of DBT (left) and DFT (right)

DBT: In DBT, objects are at first localized in each frame and then object hypotheses are linked into trajectories. Fig. 1 (left) shows the flow of DBT. Given a sequence, type-specific object detection or motion detection (based on background modeling)

[73, 74] is applied in each frame to obtain object hypotheses, then (sequential or batch) tracking is conducted to link detection hypotheses into trajectories. There are three issues worthy noting. First, in most cases object detection procedure is not the focus of DBT methods. The majority of DBT approaches build upon a pre-trained object detector which produces object hypotheses as observations. Second, as mentioned above, since object detector is trained in advance, the majority of DBT focuses on specific kinds of targets, such as pedestrians, vehicles or faces. The underlying reason is that detection of these types of objects has gained great progress in recent years [75, 76, 77]. Third, the performance of DBT depends on the performance of the employed model of object detection to a certain extent.

DFT: As shown in Fig. 1 (right), DFT [68, 78, 79, 80] requires manual initialization of a fixed number of objects in the first frame (in the form of bounding boxes or other shape configurations), then localizes these fixed number of objects in the subsequent frames. It does not rely on object detector to provide object hypotheses. Note that, when the number of objects is one, DFT degrades as the classical visual tracking problem. DBT is more popular for the fact that new objects are discovered and disappearing objects are terminated automatically. DFT requires manual initialization of each object to be tracked, thus it cannot deal with the case that objects appear. However, it is model-free, i.e., free of pre-trained object detectors. So it can deal with sequences of any type of objects. However, the setting of fixed number of objects limits its applications in practical systems.

5.2.2 Processing Mode

According to the way of processing data, MOT could be categorized into online tracking and offline tracking. The difference is whether the future frame observations are utilized when handling the current frame. Online tracking utilizes

observations up to the current time instant to conduct the estimation, while offline tracking employs observations both in the past and in the future.

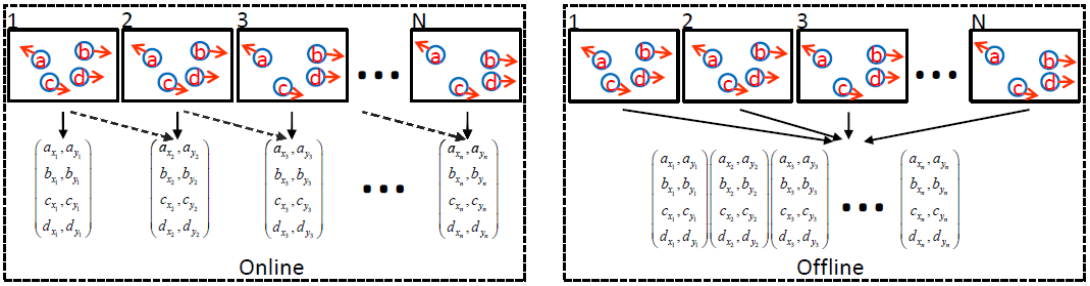

Fig. 2: Illustration of online and offline tracking.

Online tracking: In online tracking, the image sequence is handled in a step-wise way, thus online tracking is also named as sequential tracking. As shown in Figure 2 (left), we present a toy example that there are four objects (different circles) in a video sequence with IDs a, b, c and d. The arrow attached to each object indicates its movement direction. The dashed arrows represent observations in the past. The results are represented by the object's location and it's ID. Based on the up-to-time observations, trajectories are outputted on the fly.

Offline tracking: Offline tracking [74, 81, 82, 83, 43, 84, 85, 86, 87] utilizes a batch way to process the data therefore it is also called batch tracking. Figure 2 (right) illustrates how the batch tracking processes observations. Observations from all the frames are required to be obtained in advance and are investigated together to estimate the final output. Note that, due to computation ability, sometimes it is not possible to handle all the frames at one time. Alternatively, one solution is to divide the whole video into a set of segments or clips, handle these clips respectively, and infuse the results hierarchically.

5.2.3 Mathematical Methodology

MOT could be classified into probabilistic tracking and deterministic tracking according to the adopted mathematical methodology. There are two differences between them. First, the approaches to estimating states of objects are different. In probabilistic tracking, the estimation is based on probabilistic inference, while in deterministic tracking the estimation is based on deterministic optimization. Second, the outputs are different. Output of probabilistic tracking may be different in different running trials while constant in deterministic tracking.

5.3 MOT Components

As shown in Figure 3, MOT involves two primary components. One is observation model and the other one is dynamic model. Observation model measures similarity between object states and observations. To be more specific, an observation model includes modeling of appearance, motion, interaction, exclusion and occlusion. Dynamic model investigates states transition across frames. It can be classified into probabilistic inference and deterministic optimization.

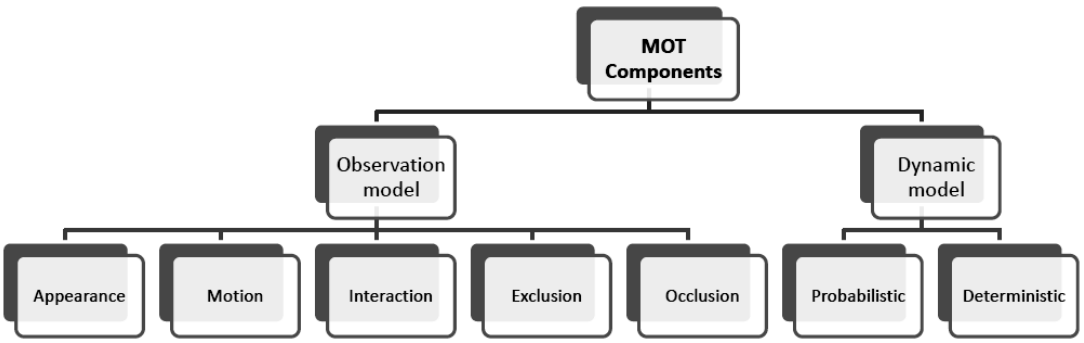

Fig. 3: Components of MOT

5.3.1 Appearance Model

Appearance is an important cue for affinity computation in MOT. However, it is worth noting that, different from single object tracking approach which primarily focuses on constructing a sophisticated appearance model to discriminate object from background, multiple object tracking does not mainly focus on appearance model, i.e., appearance cue is important but not the only cue to depend on. This is partly because that the multiple objects in MOT can hardly be discriminated by relying on only appearance information.

Technically, appearance model includes two components, i.e. visual representation and statistical measuring. Visual representation is closely related to features, but it is more than features. It is how to precisely describe the visual characteristics of object based on features, and in general it can be grouped into two sets, visual representation based on single cue and that based on multiple cues. Statistical measuring is the computation of similarity or dissimilarity between different observations when visual representation is ready.

Eq. 5.4 gives an illustration of appearance modeling, where o_i and o_j are visual representation of different observations based on single cue or multiple cues, and F (•, •) is a function to measure the similarity S_{ij} between o_i and o_j . In the following, we firstly discuss the features/cues employed in MOT, and then describe appearance models based on single cue and multiple cues respectively.

$$S_{i,j} = F(o_i, o_j) \tag{5.4}$$

Features are indispensable for MOT it was categorize into the following sub sets.

Point features: Point features are successful in single object tracking [88]. For MOT, point features can also be helpful. For instance, KLT tracker is employed to track feature points and generate a set of trajectories or short tracklets [86, 89]. Local feature points are adopted along with the bag-of-word model to capture the texture characteristics of a region.

Color/intensity features: This is the most popularly utilized feature for MOT. Usually the color or intensity features along with a measurement are employed to calculate the affinity between two counter parts (detection hypotheses, tracklets or short trajectories).

Optical flow: The optical flow feature can be employed to conduct short-term visual tracking. Thus many solutions to MOT utilize optical flow to link detection responses from continuous frames into short tracklets for further data association processing

Gradient/pixel-comparison features: There are some features based on gradient or pixel comparison. [90] utilize a variation of the level-set formula, which integrates three terms penalizing the deviation between foreground and background, an embedding function from a signed distance function and the length of the contour to track objects in continuous frames. Besides the success in human detection, HOG plays a vital role in the multiple pedestrian tracking problems as well.

Region covariance matrix features: Region covariance matrix features are robust to issues such as illumination changes, scale variations, etc. Therefore, it is also employed for the MOT problem. The region covariance matrix based dissimilarity is used to compare appearance for data association. Covariance matrices along with other features constitute the feature pool for appearance learning by [84]. [68] utilize

the covariance matrix to represent object for both single and multiple object tracking.

Depth: Depth information is employed for various computer vision tasks. With regard to MOT, Depth information is integrated into the framework to augment detection hypotheses with a depth flag 0 or 1, which further refines the detection responses. Similarly, [91] employ depth information to obtain more accurate object detections in a mobile vision system and then use the detection result for multiple object tracking.

Others: Some other features, which are not so popular, are utilized to conduct multiple object tracking as well. For instance, gait features in the frequency domain, which are unique for every person, are employed by [86] to maximize the discrimination between the tracked individuals. Given a trajectory, a line fitting via linear regression is conducted to extract the periodic component of the trajectory. Then the Fast Fourier Transform (FFT) is applied to the residual periodic signal to obtain the amplitude spectra and phase of the trajectory, which are utilized to compute the dissimilarity between a trajectory and other trajectories.

Generally speaking, most of the features are efficient. At the same time, they also have shortcomings. For instance, color histogram has well studied similarity measures, but it ignores the spatial layout of the object region. Point features are efficient, but sensitive to issues like occlusion and out-of-plane rotation. Gradient based features like HOG can describe the shape of object and robust to issues such as illumination changes, but it cannot handle occlusion and deformation well. Region covariance matrix features are more robust as they take more information in account, but this benefit is obtained at the cost of more computation. Depth features make the computation of affinity more accurate, but they require multiple views of the same scenery and/or additional algorithm to obtain depth.

Chapter-6

Proposed Method: Selection based object tracking

Our method is divided into two parts first is the preparation of background free of any foreground objects, and second part is the tracking of multiple moving objects. In the background creation part we tried to create a static background free of foreground objects from the first frame of the video.

6.1 Background creation

We have use the concept of static frame based background modeling but without creating a blank background image from start for this. Our concept is to create the background from within the video frame. For this we have chosen the first frame in the preparation of background. Then we create the frame difference between next and current frame $F_{i+1} - F_i$. We create the binary image from the frame difference image by applying threshold *Th* on the image. After creating binary image all the foreground object present in first frame (F_1 – F_0) difference is selected. Now on that selection coordinate our algorithm checks all other frame difference for no foreground object. If no foreground object found on that selection in any frame difference say $F_d = F_{i+1} - F_i$ and algorithm found no foreground object in F_d then it will copy background pixel from frame F_i to the first frame i.e. F_0 and repeat this operation till all foreground object no longer exist. And finally we get a background free of any foreground objects.

The steps of the algorithm for creating background image are given below:

Step 1: First we create an inter-frame difference

$$F_d = F_{i+1} - F_i \tag{6.1}$$

Step 2: We applied threshold to create binary image using following method.

$$B_g = \begin{cases} 1 & F_d > Th \\ 0 & F_d < Th \end{cases} \tag{6.2}$$

Step 3: Select all foreground object present in the binary image of frame ($F_1 - F_0$)

Step 4: Now from the selection list start with first selection and using that coordinate check for all other frame difference for no foreground object.

Step 5: If no foreground object is found at say F_d frame difference then we copy the background present in frame F_i ($F_d = F_{i+1} - F_i$) to the first frame F_0 , and if foreground object found in selection coordinate then we go on searching next frame.

Step 6: Again we choose next selection coordinates and repeat the search operation and when a frame difference found which have no object inside the selection boundary we copy the background to first frame, in this way finally the background will be created.

The flowchart of the background creation is given below.

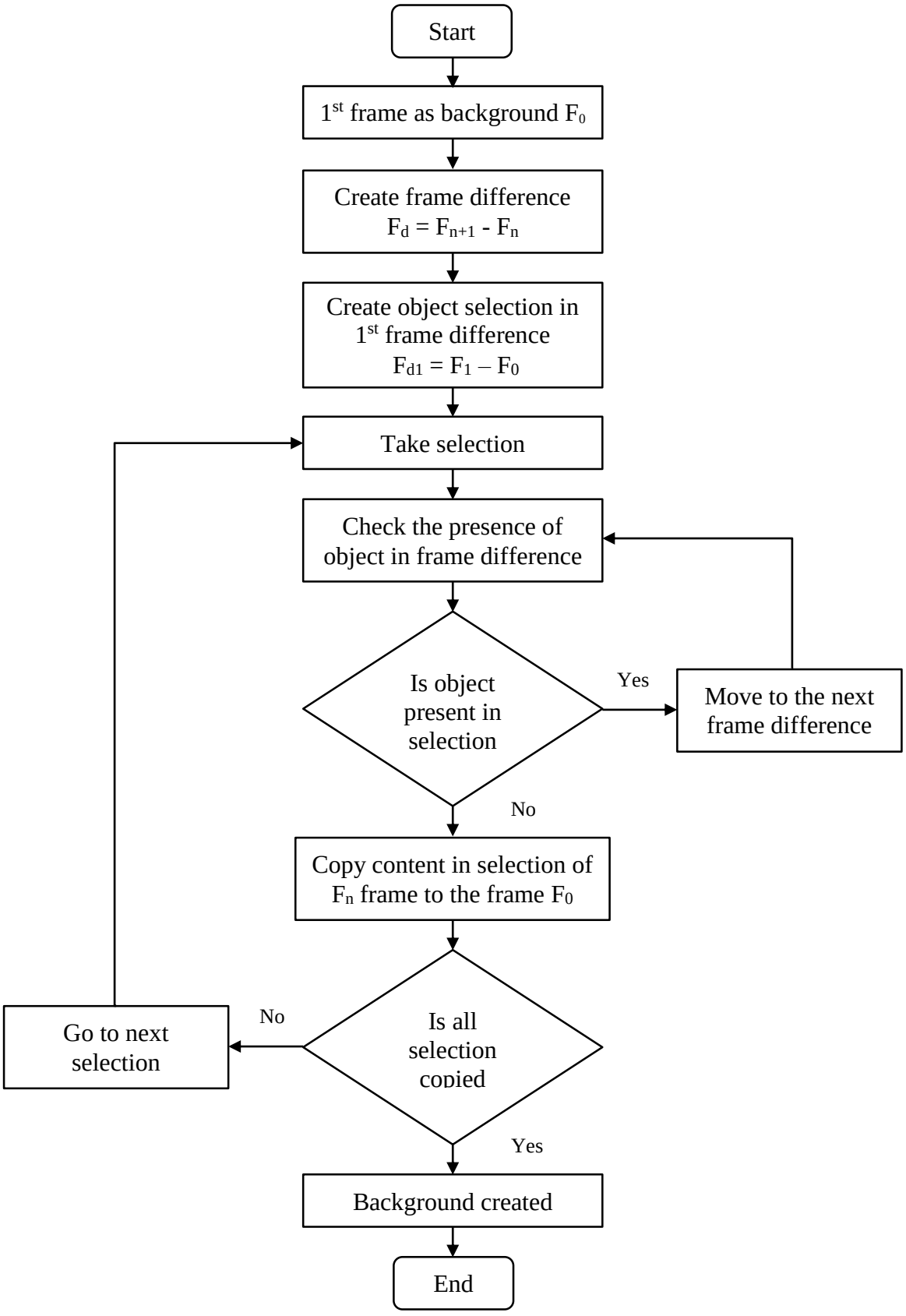

Fig. 4: Flow chart of background creating algorithm

6.2 Multi-object tracking

The multi-object tracking done by the algorithm depends mostly on proper selection of foreground objects. The selection algorithm is given in following steps and the flowchart of object tracking algorithm is given thereafter.

Step 1: We first do column scanning of binary image from top to bottom and from left to right to find first white pixel.

Step 2: When first white pixel is found we store this coordinate as left column and continue scanning for full black column which will stop our searching.

Step 3: When we found full black column we then store this coordinate as right column and start horizontal scanning inside this column.

Step 4: When we found first white pixel we store it as top coordinate for the selection, and continue scanning for full black row which will stop our searching.

Step 5: When we found the full black row we store the bottom coordinate, and in this way we get first selection coordinate of foreground object and we store it.

Step 6: We again start searching next white pixel and then black row for next selection coordinate, in this way we get the foreground objects in a column.

Step 7: When the column is finished scanning we again start vertical scanning to find next white pixel and then black column. This will again become our next boundary for searching any foreground objects.

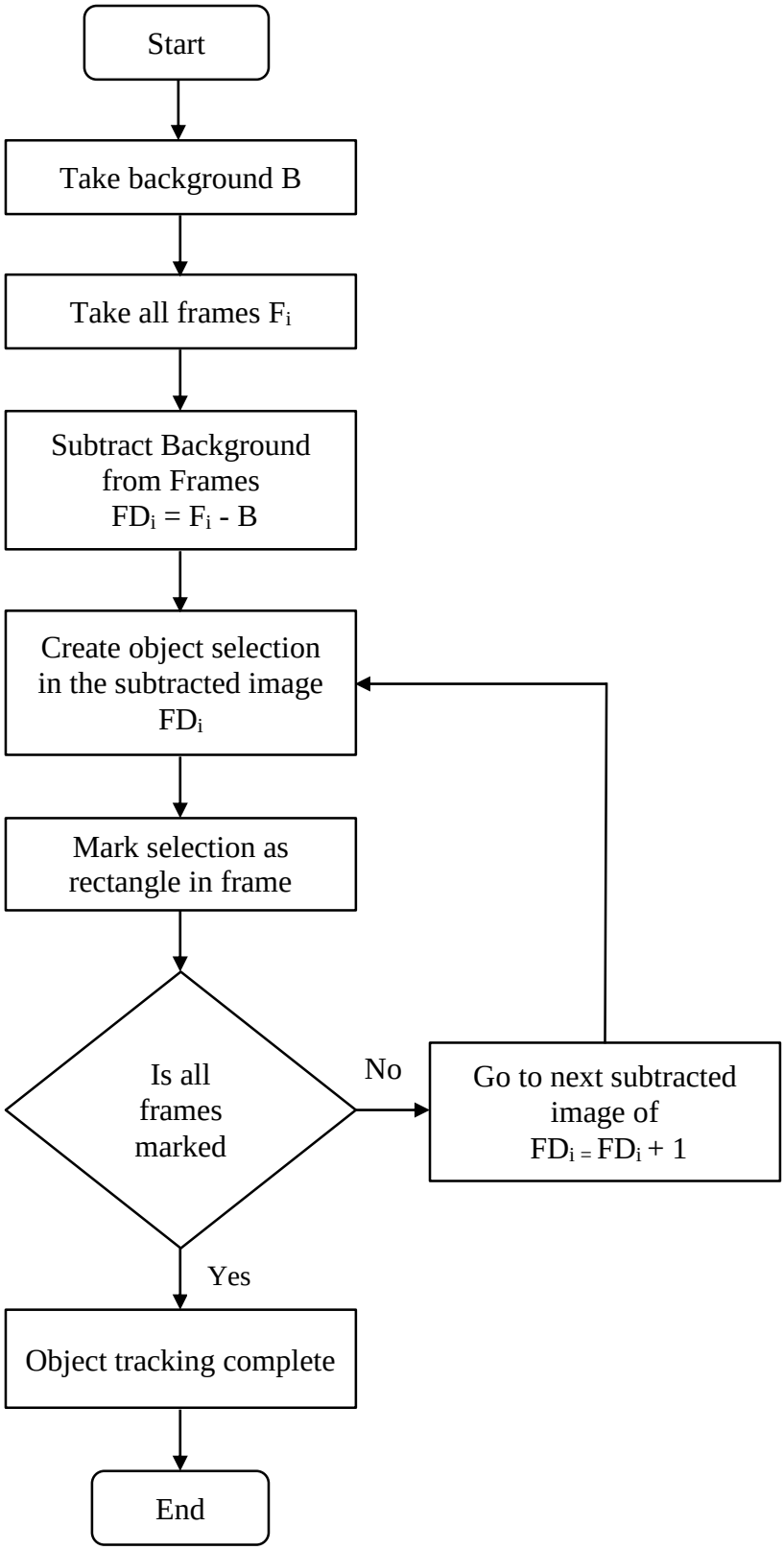

Fig. 5: Flow chart of object tracking in a video

Chapter-7

Result and Discussion

7.1 Database used in Multi-object tracking

We have used Datasets titled “EPFL data set: Multi-camera Pedestrian Videos” and are collected from website ‘http://cvlab.epfl.ch/data/pom’. We have collected 5 different categories of total 18 videos as well as two video from YouTube for testing they are titled “Sometimes Security Cameras catch a gem!” location are given below ‘www.youtube.com/watch?v=4i_GFrlaStQ’, ‘www.youtube.com/watch?v=CfAPnvGFHyM’.

All the videos at website ‘http://cvlab.epfl.ch/data/pom’ are of resolution of 360 x 288 at 25 fps, and the videos from www.youtube.com are of 480 x 360 at the rate of 30fps and 638 x 360 at 30 fps. The images used in the experiment are given below:

Fig. 6: Initial frames 1st, 54th and 100th frame of video 1

Fig.7: Initial frames 1st, 60th and 165th frame of video 2

Fig. 8: Initial frames 1st, 70th and 170th frame of video 3

Fig. 9: Initial frames 1st, 70th and 130th frame of video 4

Fig. 10: Initial frames 1st, 80th and 200th frame of video 5

Fig. 11: Initial frames 1st, 30th and 135th frame of video 6

Fig. 12: Initial frames 1st, 100th and 230th frame of video 7

Fig. 13: Initial frames 1st, 210th and 390th frame of video 8

Fig. 14: Initial frames 1st, 34th and 60th frame of video 9

7.2 Experimental Results

Here we are showing the background image created by the algorithm from 1st frame.

Fig. 15: Background of 1st, 2nd, 3rd, 4th, 5th, 6th, 7th, 8th and 9th videos

Now we are showing the tracking result of our proposed algorithm.

Fig. 16: Results of object tracking of 15th, 54th and 90th frame of video 1

Fig. 17: Result of object tracking of 15th, 55th, 70th and 170th frame of video 2

Fig. 18: Result of object tracking of 54th, 145th and 200th frame of video 3

Fig. 19: Result of object tracking of 28th, 72nd and 130th frame of video 4

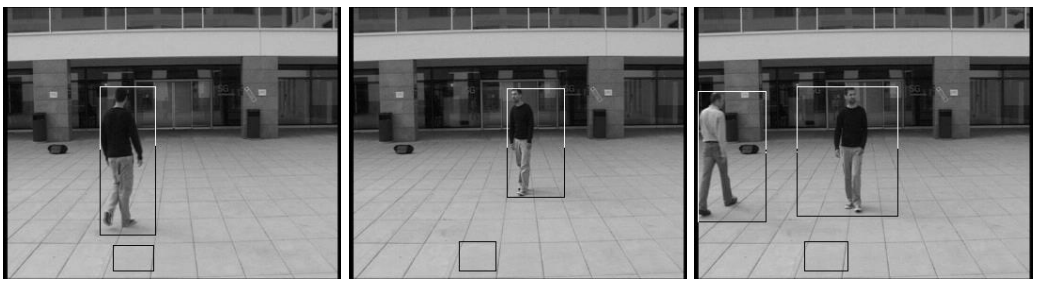

Fig. 20: Result of object tracking of 25th, 105th and 199th frame of video 5

Fig. 21: Result of object tracking of 35th, 100th and 169th frame of video 6

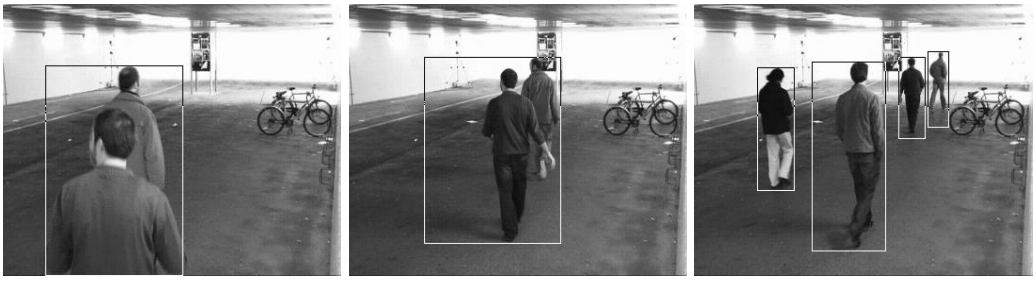

Fig. 22: Result of object tracking of 45th, 123rd and 266th frame of video 7

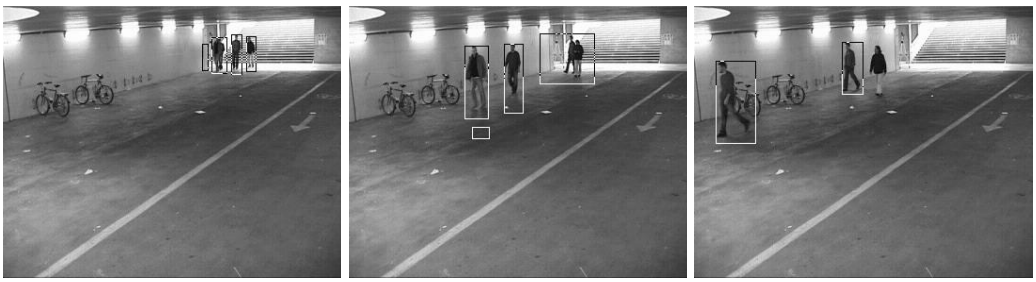

Fig. 23: Result of object tracking of 52nd, 258th and 392nd frame of video 8

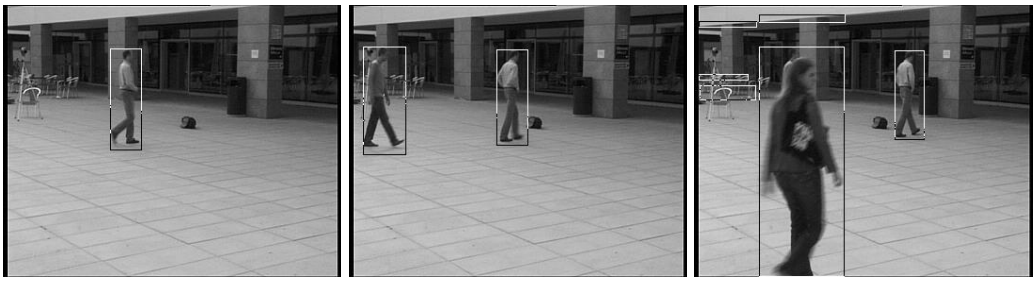

Fig. 24: Result of object tracking of 5th, 31st and 70th frame of video 9

7.3 Discussion

In this experiment of multi-object tracking we are able to detect the moving object properly by our proposed algorithms. The dataset taken some with moving object present from the beginning and some with pure background with no moving objects, to see how the algorithm handle both the situation. In the case where the moving objects present at the beginning of the video, the algorithm creates a background image which is free of any foreground object.

In the YouTube video there is lot of still image i.e. two or three of the consecutive frames are same, hence no change in frame difference obtained. Our algorithm handles this by skipping those repeated frames. Choosing threshold value between 10 to 25 gives better result. When choosing threshold value of less than 18 the shadow of object also came into consideration during background creation and it get removed.

Chapter-8

Conclusion

8.1 Summary

The task of Multi-object tracking is to locating multiple objects, maintaining their identities and results their individual path in a video sequence. Our proposed algorithm does multi-object tracking in two steps. In the first step it creates a background from the first video frame by painting the background area of the object present in other frame to the object present in the first frame by searching. After creating the background in the next step the algorithm subtract background image from the video frame to get the foreground objects, now those objects are selected and marked by a rectangle to show that tracking is going on. Our algorithm was also able to take into account the problem of repeated frames.

This algorithm works efficiently due to its low computation power requirement, and due to proper background created which is free from foreground objects.

8.2 Future work

In this work different parameters are not address such as reflection, shadow, effects of breeze, occlusion etc. another situation is not address in this work is that if an object remain still for a long time and then suddenly it moves away leaving the place vacant, that vacant area should now show into the background image, else it will show hidden object even actually that area does not have any object. That error

will propagate into selection algorithm and it will falsely produce a rectangle around nothing. In another situation if in the video no proper background was found in consecutive frames for the objects of the first frame then this technique will fail. Ours is a simple and efficient way of tracking multiple object, it can be taken as base for further improvement.

References

[1] A. Criminisi, P. Perez, and K. Toyama, “Object removal by exemplar-based inpainting,” Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 721-728, June 2003.

[2] A. Kokaram, B. Collis and S. Robinson, “A Bayesian framework for recursive object removal in movie postproduction,” IEEE International Conference. on Image Processing, vol. 1, pp. 937-940, September 2003.

[3] O. Rostamianfar, F. Janabi-Sharifi, and I. Hassanzadeh, “Visual tracking system for dense traffic intersections,” Canadian Conference on Electrical and Computer Engineering, pp. 2000–2004, May 2006.

[4] K.A. Patwardhan, G. Sapiro, and M. Bertalmio, “Video inpainting under constrained camera motion,” IEEE Transactions on Image Processing, vol. 16, issue 2, pp. 545–553, February 2007.

[5] S.-C. Cheung and C. Kamath, “Robust techniques for background subtraction in urban traffic video,” Proceedings of the SPIE, vol. 5308, pp. 881-892, 2004.

[6] A.-N. Lai, H. Yoon, and G. Lee, “Robust background extraction scheme using histogram-wise for real-time tracking in urban traffic video,” IEEE Conference on Computer and Information Technology, pp. 845-850, July 2008.

[7] Srenivas Varadarajan, Lina J. Karam, and Dinei Florencio, “Background recovery from video sequences using motion parameters”, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 989-992, 2009

[8] M. Saska, J. Chudoba, L. Přeučil, J. Thomas, G. Loianno, A. Třešňák, V. Vonásek, V. Kumar, “Autonomous Deployment of Swarms of Micro-Aerial Vehicles in Cooperative Surveillance.”, In Proceedings of International Conference on Unmanned Aircraft System (ICUAS), pp. 584-595, May 2014

[9] M. Saska, J. Chudoba, L. Přeučil, “Swarms of Micro Aerial Vehicles Stabilized Under a Visual Relative Localization”, In Proceedings of IEEE International Conference on Robotics and Automation (ICRA), pp. 3570-3575, May 2014

[10] B.A. Boghossian, S.A. Velastin, “Image processing system for pedestrian monitoring using neural classification of normal motion patterns,” Measurement and Control, vol. 32, issue 9, pp. 261-264, 1999.

[11] B.A. Boghossian, S.A. Velastin, “Motion-based machine vision techniques for the management of large crowds.” In Proceedings of IEEE 6th International Conference on Electronics, Circuits and Systems, pp. 5-8, September 1999.

[12] F. Galgani, Sun Yiwen, P.L. Lanzi and J. Leigh, “Automatic analysis of eye tracking data for medical diagnosis”, In Proceeding of IEEE Symposium on Computational intelligence and Data Mining, pp. 195-202, March 2009

[13] Yi Li, Songde Ma, Hanqing Lu, “Human posture recognition using multi-scale morphological method and Kalman motion estimation.”, In Proceeding of IEEE International Conference on Pattern Recognition, pp. 175-177, 1998.

[14] J. Segen, S. Kumar, “Shadow gestures: 3D hand pose estimation using a single camera”, In Proceeding of IEEE CS Conference on Computer Vision and Pattern Recognition, pp. 479-485, 1999.

[15] M. Turk, “Visual interaction with life like characters.”, In Proceeding of IEEE International Conference on Automatic Face and Gesture Recognition, Killington, pp. 368-373, 1996.

[16] Cem Hocaoğlu and Arthur C. Sanderson, “Planning Multiple Paths with Evolutionary Speciation”, IEEE Transactions on Evolutionary Computation, vol. 5, issue 3, pp. 169-191, June 2001

[17] Horesh Ben Shitrit1, Jerome Berclaz, Francois Fleuret and Pascal Fua “Tracking Multiple People under Global Appearance Constraints”, IEEE International Conference on Computer Vision (ICCV), pp. 137-144, 2011.

[18] Isaac Cohen and Gerard Medioni “Detecting and Tracking Moving Objects for Video Surveillance”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 319-325, 1999

[19] Kuihe Yang, Zhiming Cai, Lingling Zhao, “Algorithm Research on Moving Object Detection of Surveillance Video Sequence”, Optics and Photonics Journal, vol. 3, pp. 308-312, June 2013

[20] Muyun Weng, Guoce Huang and Xinyu Da, “A New Interframe Difference Algorithm for Moving Target Detection”, 3rd International Congress on Image and Signal Processing (CISP), pp. 285-289, October 2010

[21] François Brémond and Monique Thonnat, “Tracking Multiple Nonrigid Objects in Video Sequences”, IEEE Transactions on Circuits and Systems for Video Technology, vol. 8, no. 5, pp. 585-591, September 1998

[22] Hyungki Roh, Seonghoon Kang and Seong-Whan Lee, “Multiple People Tracking Using an Appearance Model Based on Temporal Color”, Proceedings of the 1st IEEE International Workshop in Biologically Motivated Computer Vision (BMCV), vol. 1811 pp. 369-378, May 2000

[23] Yoshinori Ohno, Jun Miura, and Yoshiaki Shirai, “Tracking Players and Estimation of the 3D Position of a Ball in Soccer Games”, Proceedings of the 15th International Conference on Pattern Recognition, vol. 1, pp. 145-148, September 2000

[24] Sohaib Khan, Omar Javed, Zeeshan Rasheed, Mubarak Shah, “Human Tracking in Multiple Cameras”, Proceedings of 8th IEEE International Conference on Computer Vision (ICCV), vol. 1, pp. 331-336, July 2001

[25] Koichi Sato and J. K. Aggarwal, “Tracking and Recognizing Two-person Interactions in Outdoor Image Sequences”, Proceedings of IEEE Workshop on Multi-Object Tracking, pp. 87-94, 2001

[26] Shiloh L. Dockstadert and A. Murat Tekalp, “Multiple Camera Fusion for Multi-Object Tracking”, Proceedings of IEEE Workshop on Multi-Object Tracking, pp. 95-102, 2001

[27] Dieter Koller, Joseph Weber and Jitendra Malik, “Robust Multiple Car Tracking with Occlusion Reasoning”, Proceedings of 3rd European Conference on Computer Vision, vol. 1, pp. 189-196, May 1994

[28] A. Criminisi, P. Perez, and K. Toyama, “Object removal by exemplar-based inpainting”, Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 721 -728, June 2003

[29] A. Kokaram, B. Collis and S. Robinson, “A Bayesian framework for recursive object removal in movie postproduction”, IEEE International Conference on Image Processing, vol. 1, pp. 937-40, September 2003

[30] O. Rostamianfar, F. Janabi-Sharifi, and I. Hassanzadeh, “Visual tracking system for dense traffic intersections”, Canadian Conference on Electrical and Computer Engineering, pp. 2000–2004, May 2006

[31] A.-N. Lai, H. Yoon, and G. Lee, “Robust background extraction scheme using histogram-wise for real-time tracking in urban traffic video”, IEEE Conference on Computer and Information Technology, pp. 845-850, July 2008

[32] H.-J. Hsu, J.-F. Wang, and S.-C. Liao, “A hybrid algorithm with artifact detection mechanism for region filling after object removal from a digital photograph”, IEEE Transactions on Image Processing, vol. 16, issue 6, pp. 1611–1622, June 2007

[33] C. Herley, “Automatic occlusion removal from minimum number of images”, IEEE International Conference on Image Processing, vol. 2, pp. 1046-1049, September 2005

[34] P.M.Q. Aguiar and J.M.F. Moura, “Joint segmentation of moving object and estimation of background in low-light video using relaxation”, IEEE International Conference on Image Processing, vol. 5, pp. 53-56, September 2007

[35] Y. Lu, W. Ga, and F. Wu, “Automatic video segmentation using a novel background model”, IEEE International Symposium on Circuits and Systems, vol. 3, pp. 807–810, May 2002

[36] Y. Zhang, J. Xiao, and M. Shah, “Motion layer based object removal in videos”, 7th IEEE workshop on Application of Computer Vision, vol. 1, pp. 516-521, January 2005

[37] F.-F. Meng, Z.-S. Qu, and Q.-S. Zeng, and L. Li, “Traffic object tracking based on increased-step motion history image”, IEEE International Conference on Automation and Logistics, pp. 345–349, August 2007

[38] S. Kang, J. Paik, A. Koschan, B. Abidi, and M. A. Abidi (2003). “Real-time video tracking using PTZ cameras”, Procedings of SPIE, vol. 5132, pp. 103–111, 2003.

[39] D. Comaniciu, V. Ramesh, P. Meer, “Real-time tracking of non-rigid objects using mean shift”, Proceedings. IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 142-149, 2000

[40] James Black, Tim Ellis, and Paul L. Rosin, “A Novel Method for Video Tracking Performance Evaluation”, Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS), pp. 125–132, December 2003

[41] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking”, IEEE Transactions on Signal Processing, vol. 50, issue 2, pp. 174-188, February 2002

[42] J. Martinez-del-Rincon, D. Makris, C. Orrite-Urunuela and J.-C. Nebel, “Tracking Human Position and Lower Body Parts Using Kalman and Particle Filters Constrained by Human Biomechanics”, IEEE Transactions on Systems Man and Cybernetics - Part B(Cybernetics), vol. 41, issue 1, pp. 26-37, April 2010

[43] Yang B, Huang C, Nevatia R, “Learning affnities and dependencies for multi-target tracking using a CRF model”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 1233-1240, 2011

[44] Pellegrini S, Ess A, Schindler K, Van Gool L, “You'll never walk alone: Modeling social behavior for multitarget tracking”, In Proceedings of IEEE International Conference Computer Vision, pp. 261-268, 2009

[45] Koller D, Weber J, Malik J, “Robust multiple car tracking with occlusion reasoning”, In Proceedings of European Conference Computer Vision, pp. 189-196, 1994

[46] Betke M, Haritaoglu E, Davis LS, “Real-time multiple vehicle detection and tracking from a moving vehicle”, Machine Vision Application, vol. 12, issue 2, pp. 69-83, 2000

[47] Lu WL, Ting JA, Little JJ, Murphy KP, “Learning to track and identify players from broadcast sports videos”, IEEE Transaction Pattern Analysis Machine Intelligence, vol. 35, issue 7, pp. 1704-1716, 2013

[48] Xing J, Ai H, Liu L, Lao S, “Multiple player tracking in sports video: a dual-mode two-way bayesian inference approach with progressive observation modeling”, IEEE Transaction Image Processing, vol. 20, issue 6, pp. 1652-1667, 2011

[49] Nillius P, Sullivan J, Carlsson S, “Multi-target trackinglinking identities using bayesian network inference”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 2187-2194, 2006

[50] Luo W, Kim TK, Stenger B, Zhao X, Cipolla R, “Bilabel propagation for generic multiple object tracking”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 1290-1297, 2014

[51] Betke M, Hirsh DE, Bagchi A, Hristov NI, Makris NC, Kunz TH, “Tracking large variable numbers of objects in clutter”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 1-8, 2007

[52] Khan Z, Balch T, Dellaert F, “An mcmc-based particle filter for tracking multiple interacting targets”, In Proceedings of European Conference Computer Vision, pp. 279-290, 2004

[53] Spampinato C, Chen-Burger YH, Nadarajan G, Fisher RB, “Detecting, tracking and counting fish in low quality unconstrained underwater videos”, Proceedings of International Conference Computer Vision Theory Application, pp. 514-519, 2008

[54] Fontaine E, Barr AH, Burdick JW, “Model-based tracking of multiple worms and fish”, In Proceedings of IEEE International Conference Computer Vision Workshops, pp. 1-13, 2007

[55] Meijering E, Dzyubachyk O, Smal I, van Cappellen WA, “Tracking in cell and developmental biology”, Seminar Cell Development Biology, vol. 20, issue 8, pp. 894-902, 2009

[56] Li K, Miller ED, Chen M, Kanade T,Weiss LE, Campbell PG, “Cell population tracking and lineage construction with spatiotemporal context”, Medical Image Analysis, vol. 12, issue 5, pp. 546-566, 2008

[57] Liu Y, Li H, Chen YQ, “Automatic tracking of a large number of moving targets in 3D”, In Proceedings of European Conference Computer Vision, pp. 730-742, 2012

[58] Breitenstein MD, Reichlin F, Leibe B, Koller-Meier E, Van Gool L, “Robust tracking-by-detection using a detector confidence particle filter”, In Proceedings of IEEE International Conference Computer Vision, pp. 1515-1522, 2009

[59] Yang M, Lv F, Xu W, Gong Y, “Detection driven adaptive multi-cue integration for multiple human tracking”, In Proceedings of IEEE International Conference Computer Vision, pp. 1554-1561, 2009

[60] Mitzel D, Leibe B, “Real-time multi-person tracking with detector assisted structure propagation”, In Proceedings of IEEE International Conference Computer Vision Workshops, pp. 974-981, 2011

[61] Rodriguez M, Sivic J, Laptev I, Audibert JY, “Datadriven crowd analysis in videos”, In Proceedings of IEEE International Conference Computer Vision, pp. 1235-1242, 2011

[62] Kratz L, Nishino K, “Tracking with local spatiotemporal motion patterns in extremely crowded scenes”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 693-700, 2010

[63] Reid DB, “An algorithm for tracking multiple targets”, IEEE Transaction Automation Control, vol. 24, issue 6, pp. 843-854, 1979

[64] Jin Y, Mokhtarian F, “Variational particle filter for multi-object tracking”, In Proceedings of IEEE International Conference Computer Vision, pp. 1-8, 2007

[65] Yang C, Duraiswami R, Davis L, “Fast multiple object tracking via a hierarchical particle filter”, In Proceedings of IEEE International Conference Computer Vision, pp. 212-219, 2005

[66] Hess R, Fern A, “Discriminatively trained particle filters for complex multi-object tracking”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 240-247, 2009

[67] Han B, Joo SW, Davis LS, “Probabilistic fusion tracking using mixture kernel-based bayesian filtering”, In Proceedings of IEEE International Conference Computer Vision, pp. 1-8, 2007

[68] Hu W, Li X, Luo W, Zhang X, Maybank S, Zhang Z, “Single and multiple object tracking using log-euclidean riemannian subspace and block-division appearance model”, IEEE Transaction Pattern Analysis Machine Intelligence, vol. 34, issue 12, pp. 2420-2440, 2012

[69] Khan Z, Balch T, Dellaert F, “Mcmc-based particle filtering for tracking a variable number of interacting targets”, IEEE Transaction Pattern Analysis Machine Intelligence, vol. 27, issue 11, pp. 1805-1819, 2005

[70] Khan Z, Balch T, Dellaert F, “Mcmc data association and sparse factorization updating for real time multitarget tracking with merged and multiple measurements”, IEEE Transaction Pattern Analysis Machine Intelligence, vol. 28, issue 12, pp. 1960-1972, 2006

[71] Choi W, Pantofaru C, Savarese S, “A general framework for tracking multiple people from a moving camera”, IEEE Transaction Pattern Analysis Machine Intelligence, vol. 35, issue 7, pp. 1577-1591, 2013

[72] Yang B, Nevatia R, “Online learned discriminative part-based appearance models for multi-human tracking” In Proceedings of European Conference Computer Vision, pp. 484-498, 2012

[73] Bose B, Wang X, Grimson E, “Multi-class object tracking algorithm that handles fragmentation and grouping” In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 1-8, 2007

[74] Song B, Jeng TY, Staudt E, Roy-Chowdhury AK, “A stochastic graph evolution framework for robust multitarget tracking”, In Proceedings of European Conference Computer Vision, pp. 605-619, 2010

[75] Dalal N, Triggs B, “Histograms of oriented gradients for human detection”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 886-893, 2005

[76] Felzenszwalb P, Girshick R, McAllester D, Ramanan D, “Object detection with discriminatively trained part-based models”, IEEE Transaction Pattern Analysis Machine Intelligence, vol. 32, issue 9, pp. 1627-1645, 2010

[77] Sun Z, Bebis G, Miller R, “On-road vehicle detection: A review”, IEEE Transaction Pattern Analysis Machine Intelligence, vol. 28, issue 5, pp. 694-711, 2006

[78] Zhang L, van der Maaten L, “Structure preserving object tracking”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 1838-1845, 2013

[79] Zhang L, van der Maaten L, “Preserving structure in model-free tracking”, IEEE Transaction Pattern Analysis Machine Intelligence, vol. 36, issue 4, pp. 756-769, 2014

[80] Yang M, Yu T, Wu Y, “Game-theoretic multiple target tracking”, In Proceedings of IEEE International Conference Computer Vision, pp. 1-8, 2007

[81] Qin Z, Shelton CR, “Improving multi-target tracking via social grouping”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 1972-1978, 2012

[82] Yang B, Nevatia R, “Multi-target tracking by online learning of non-linear motion patterns and robust appearance models”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 1918-1925, 2012

[83] Brendel W, Amer M, Todorovic S, “Multiobject tracking as maximum weight independent set”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 1273-1280, 2011

[84] Kuo CH, Huang C, Nevatia R, “Multi-target tracking by on-line learned discriminative appearance models”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 685-692, 2010

[85] Henriques JF, Caseiro R, Batista J, “Globally optimal solution to multi-object tracking with merged measurements”, In Proceedings of IEEE International Conference Computer Vision, pp. 2470-2477, 2011

[86] Sugimura D, Kitani KM, Okabe T, Sato Y, Sugimoto A, “Using individuality to track individuals: clustering individual trajectories in crowds using local appearance and frequency trait”, In Proceedings of IEEE International Conference Computer Vision, pp. 1467-1474, 2009

[87] Choi W, Savarese S, “Multiple target tracking in world coordinate with single, minimally calibrated camera”, In Proceedings of European Conference Computer Vision, pp. 553-567, 2010

[88] Shi J, Tomasi C, “Good features to track”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 593-600, 1994

[89] Zhao X, Gong D, Medioni G, “Tracking using motion patterns for very crowded scenes”, In Proceedings of European Conference Computer Vision, pp. 315-328, 2012

[90] Mitzel D, Horbert E, Ess A, Leibe B, “Multi-person tracking with sparse detection and continuous segmentation”, In Proceedings of European Conference Computer Vision, pp. 397-410, 2010

[91] Ess A, Leibe B, Schindler K, Van Gool L, “A mobile vision system for robust multi-person tracking”, In Proceedings of IEEE International Conference Computer Vision Pattern Recognition, pp. 1-8, 2008