Behavior Understanding in Non-Crowded and Crowded Scenes

Dissertation

zur Erlangung des akademischen Grades

Doktoringenieurin

(Dr.-Ing.)

von M.Eng. Saira Saleem Pathan

geb. am 27.01.1980 in Hyderabad Pakistan

genehmigt durch die Fakultät für Elektrotechnik und Informationstechnik

der Otto-von-Guericke-Universität Magdeburg

Gutachter:

Prof. Dr.-Ing. habil. Bernd Michaelis
Prof. Dr. rer. nat. Christian Wöhler
J.-Prof. Dr.-Ing. habil. Ayoub Al-Hamadi

Promotionskolloquium am 18.06.2012

Acknowledgments

I would like to thank my supervisor Prof. Dr.-Ing. habil. Bernd Michaelis for his guidance and relentless support over the course of this PhD. His commitment of research inspires me over all these years and motivates me to do my best. I am grateful to Prof. Dr. rer. nat. Christian Wöhler and J.-Prof. Dr.-Ing. habil. Ayoub Al-Hamadi for accepting to review my thesis. Thanks to Prof. Dr.-Ing. Christian Diedrich and Prof. Dr.-Ing. Roberto Leidhold for being in the examination committee.

I would like to thank my colleagues in AGMI research group, particularly, Dr. Gerald Krell, Omer Rashid and Axel Panning for their countless stimulating conversation and invaluable technical advices. Special thanks to my family for their infinite support and understanding.

Abstract

This dissertation presents a novel approach to the problem of object tracking and behavior understanding for non-crowded and crowded scenes within computer vision. In particular for non-crowded scenes, this dissertation contributes to both tracking and behavior understanding components of a surveillance system. We propose an integrated top to down framework and incorporate the qualitative modeling of human perception with the quantitative approach. In quantitative approach, a Bayesian Matching Weight is devised to measure the matching weights among objects by exploiting Color Structure Code approach and Ellipse Histogram as object representative features. In qualitative approach, we propose the axioms which are not based on statistical measures of typicality, but upon building an understanding of the way people percept the behaviors in tracking scenarios. Tracking of the objects is achieved by employing Kalman filter for localization in which every object is modeled as a linear system. We have tested the robustness of proposed approach on three benchmark datasets and have verified qualitatively the performance.

In particular for crowded scenes, we propose a top to down framework to understand the crowd behaviors and analyze the anomaly by modeling the crowd dynamics in video sequences. We section the video sequence into spatio-temporal regions named as flow-block which allows the marginalization of arbitrary flow cloud data computed by optical flow. We apply the Social Entropy to address the issues of optical flow noise and empirically determine a quantitative metric to generate the refined flow cloud data. We employ mixture of Gaussians for modeling flow cloud data to generate the flow patterns (flow features) in each flow-block. In the context of characterization of crowd behaviors, we employ two classification approaches named as Support Vector Machines and Conditional Random Field. We have tested the robustness of the proposed approach on two benchmark datasets and have verified qualitatively the performance. Moreover, we have achieved 97% classification rate which is dominating when compared to state of the art approaches.

Zusammenfassung

Diese Dissertation behandelt einen neuen Lösungsansatz für das Problem der Verhaltensanalyse von Szenen mit dicht gedrängten Personengruppen und Szenen mit wenigen Personen durch Methoden der Computer Vision. Für Szenen mit wenigen Personen leistet diese Dissertation einen Beitrag sowohl für das Tracking als auch für Komponenten zur Verhaltensanalyse von Überwachungssystemen. Es wird ein integrierter Top-Down-Ansatz vorgeschlagen, der die qualitative Modellierung der menschlichen Wahrnehmung als auch den quantitativen Ansatz beinhaltet. Im quantitativen Ansatz wurde ein Bayes Matching Gewicht genutzt, um die Gewichte des Objektmatchings zu berechnen. Als Objektmerkmale kamen hier der Color Structure Code sowie Ellipsen Farbhistogramme zum Einsatz. Im qualitativen Ansatz werden Axiome verwendet, die nicht auf statistischen Typizitätswerten beruhen, sondern darauf, wie Menschen das Verhalten in Szenen wahrnehmen. Das Objekttracking zur Lokalisierung der Personen wird durch Hinzunahme eines Kalmanfilters erreicht, welcher für jedes Objekt ein lineares System modelliert. Die Robustheit des vorgeschlagenen Ansatzes wurde auf drei Benchmark-Datensätzen getestet und qualitativ validiert.

Insbesondere für Szenen mit vielen dicht gedrängten Personen wird ein Top-Down-Ansatz vorgeschlagen um das Verhalten von grossen Menschenmengen zu verstehen und um Anomalien zu analysieren, indem die Menschenmenge dynamisch modelliert wird. Die Videosequenz wird dazu in räumlich, zeitliche Abschnitte unterteilt, die Flussblöcke genannt werden, um das mittels optischem Fluss bestimmte Flussfeld zu vereinfachen. Das Problem des Rauschens im optischen Fluss wird durch die Anwendung von sozialer Entropie behandelt. Hierfür wurde empirisch eine quantitative Metrik bestimmt, mit deren Hilfe verbesserte Flussfelder generiert werden. Um Muster innerhalb der Flussblöcke zu bestimmen, werden Gauss-Mixtur-Modelle verwendet. Diese repräsentieren die Charakteristik der zu Grunde liegenden Dynamik. Im Rahmen der Untersuchungen zum Verhalten von gedrängten Menschenmengen wurden zwei Methoden verwendet: Support Vector Machines und Conditional Random Fields. Die Leistungsfähigkeit beider Klassifikationsschemata wurde anhand zweier Benchmark-Datensätzen quantitativ und qualitativ verglichen. Darüber wurden in den Ergebnisse Klassifizierungsraten von 97% erreicht, womit die Ergebnisse aus dem state-of-art übertroffen wurden.

Dedication

To my grand parents for envisioning their dreams in me,
To my family for supporting in the persuasion of their dreams, and
To my husband Omer for being my motivation and life.

Declaration

I, hereby declare that the presented work is done without undue assistance from third parties and have made no use other than the indicated resources directly or indirectly.
Some parts of the work presented in this thesis have been published in international conferences and journals (in publication list).

Magdeburg, 22. Feb 2012

Saira Saleem Pathan

Contents

List of Figures

List of Tables

List of Acronyms

ABMM Adaptive Background Mixtures Model.

AI Artificial Intelligence.

AMF Approximated Median Filter.

BMW Bayesian Matching Weight.

CA Cellular Automata.

CRF Conditional Random Field.

CSC Color Structure Code.

CTM Correlated Topic Model.

DpT Detection prior to Tracking.

EM Expectation Maximization.

GPU Graphical Processor Unit.

HMM Hidden Markov Model.

HOF Histogram of Flow.

JPDA Joint Probability Data Association.

KL Kullback-Leibler.

LDA Linear Discriminant Analysis.

MDI Median Difference Image.

MHT Multiple Hypothesis Tracking.

PCA Principal Component Analysis.

PETS2006 Performance Evaluation of Tracking and Surveillance 2006.

PETS2009 Performance Evaluation of Tracking and Surveillance 2009.

RGA Running Gaussian Average.

SE Social Entropy.

SFM Social Force Model.

SVM Support Vector Machine.

TpD Tracking prior to Detection.

List of Formula Symbols

ε_h	normalized ellipse color histogram
$\hat{\sigma}^2$	estimate of variance of Gaussian distribution
$\hat{\vec{\mu}}$	estimate of mean value of Gaussian distribution
$\hat{w}$	non-negative weights of Gaussian distributions which are summed up to one.
$\vec{f}_p$	flow cloud data in each flow-blocks in which each flow field contains flow velocities $\vec{f}_p = (v_x, v_y)$ where v_x and v_y represent the velocities along with horizontal and vertical axis of the motion field.
ζ_p	color-patches of object computed with CSC approach
$area$	area of object
bb	bounded region around the object
$c_n^{k:id}$	color-patch of object with unique id with a set of attributes such as area of color-patch c_{area}, mean color of the color-patch $c_{\overline{ncolor_{rgb}}}$, and the bounding region of the color-patch c_{bb}.

$D \in [0,1]$ KL divergence between objects at $I(k-1)$ and $I(k)$

e_{active} exit tracking event which is set to $true$ if any object leave the scene.

$F_{(m,l)}$ flow-blocks for each video segments S_n

f_{feat} feature set computed for each detected objects

$I(k)$ image at time k

$I(k-1)$ image at time $k-1$

n_{active} new tracking event which is set to $true$ if any new object is entered in the scene.

o_j^{id} the m detected objects with unique identities id at $I(k)$. Each detected object contains a set of features in $o_j^{id} = (id, w_m, Q_q, Q_l)$.

o_{active} occlusion tracking event which is set to $true$ when two or more objects occlude each other.

Q_l set of behavioral states of objects (N, Oc, Ov, R, Ne, E) where N is the normal behavioral state, Oc is the occluded behavioral state, Ov is overlaper behavioral, R is the reappeared object behavioral state, Ne is the new object behavioral state, and E is the exit object behavioral state.

Q_q quantitative characteristics of objects $(\varepsilon_h, \zeta_p, area, bb)$

s_{active} split tracking event which is set to $true$ when two or more objects are separated from occlusion.

S_n n overlapping video segments of given video

S_r relative spatial occupancy and determines the relationship among the objects at k and $k-1$.

V_{th} empirical threshold for background subtraction

$w(o_i^{id})_{csc}$ CSC-Based Object Prior-Weight computed for each object at $I(k-1)$

w_m matching weight of the object

x_t pixel history in image sequence

CHAPTER 1

Introduction

The human visual system perceives, interprets and understands the multidimensional structure of the world with apparent ease. Researchers from the field of psychology, sociology, civil engineering, computer graphics, and computer vision have spent decades in trying to discover the secrets of how the visual system works. Particularly, in computer vision, understanding the behavior of objects has gained an increasing attention because it directly interfaces and simulates the visual information. Besides, it deals with a wide range of applications, such as surveillance in non-crowded scenes, and situation analysis in crowded scenes.

In this dissertation, we are interested in understanding the behavior of objects, such as where they are, what they are doing, and their collective interactions. Our goal is to make computers, understand the object behaviors in the visual world as the human perceives it which is valuable in security and commercial applications for both non-crowded and crowded scenes. The non-crowded scene includes a sparse number of people typically in a unruly way as shown in Figure 1.1(a). In contrast, the crowded scene defines a gathering of large number of people together, typically in an organized or disorganized way as shown in Figure 1.1(b). We have developed methods that address some of the critical aspects of understanding the object behaviors for visual scenes in two aspects. For non-crowded scenes, the behavior understanding is to discriminate varied object behaviors (e.g., normal walk, occluded, leaving the scene, entering the scene, speed, and orientation of the walk.) over time in the scene through object tracking. In contrast, the behavior analysis and understanding in crowded scenes are to characterize various crowd behaviors (e.g., normal walk, running, dispersion, or abnormal behavior) over time.

Although, we are analyzing the behaviors in both non-crowded and crowded scenes, but we have developed a generic framework for low level processing to perform object detection and visual feature computation. As the goal of behavior understanding of non-crowded and crowded scenes are quite different so, we have tackled these goals separately. Therefore, two frameworks are proposed to achieve the objectives of behavior understanding in non-crowded and crowded scenes.

a) b)

Figure 1.1: shows an example of visual scenes from PETS2006 [1] and PETS2009 [3] datasets. a) describes the scene containing few people moving with random aims. b) shows a crowded scene where people are moving in groups with some defined and undefined aims.

1.1 Motivation

An ideal behavior understanding system should be able to detect all entities, such as people and meaningful objects in the monitored scene, track them over time, and infer all the relationships among them. In the non-crowded scenes, these systems should be capable to easily detect the individual and collective activities like running, group walk, or excessive loitering. In contrast, for the crowded scenes, it should be able to characterize and detect large scale events like crowd formation, panic, or abnormality patterns.

1.1.1 Non-crowded Scenes

Behavior understanding of objects in non-crowded scenes includes the basic components, such as tracking of objects, and inference about their individual (e.g., normal walk, occlusion, speed and orientation of the walk) and collective behaviors (e.g., when two objects occlude, which object occludes the other object, or when the objects split).

We are living in a surveillance environment where the security cameras are installed at locations, such as train stations, subways, corridors, shops, and foyers which are providing continuous video streams. The interested commodities can utilize these videos and analyze the range of events and activities associated with object behaviors that can be tracked and detected. For instance, security officials might be interested in analyzing the behaviors of objects in the underlying scene and to keep an eye on their activities. An example scenario from PETS2006 [1] is

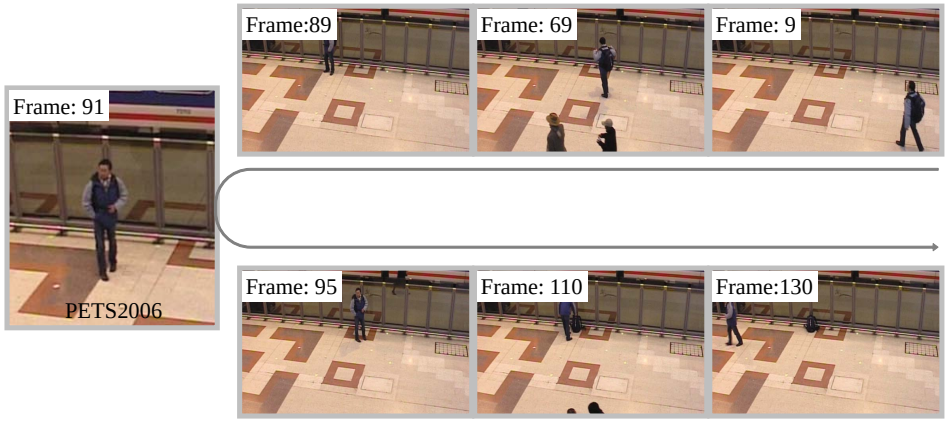


Figure 1.2: shows an example from PETS2006 [1] dataset where behaviors of objects are highlighted in *Frames 9-130*. It is noticed that object (i.e., highlighted in *Frame 9*) enters in scene and keeps on strolling unlike other fellow pedestrians.

demonstrated in Figure 1.2, where the majority of people are walking across the corridor of the train station, but a person is strolling suspiciously in the corridor. In such public places, it is quite common to loose the track of objects due to severe occlusion arising from the interaction of objects.

Therefore, it is crucial to have a framework that can help in overcoming these difficulties by employing vision methods that focus on extracting all sources of information and detecting the object behaviors in non-crowded scenes. The phenomenon of these methods should be synthesized with mathematical, inferential, and statistical models. Moreover, the behavior understanding system for non-crowded scenes should be able to perform persistent object tracking, infer, and understand the semantics of their behaviors, such as normal walk, occluded object, overlapping object, new and exit object over time. Further, it should be able to cop many real-time situations containing issues, such as objects are not traceable due to the failure of detection algorithm, fragmentation of objects, clutter due to occlusion, and uncertainty due to noisy visual clues. These issues make object behavior understanding tasks, such as detection, tracking, and behavior inferencing challenging which are the fundamental components of surveillance systems.

1.1.2 Crowded Scenes

Behavior understanding in crowded scenes at locations, such as city centers, social, and religious gathering of people, is an emerging research domain in computer vi-

Figure 1.3: shows an example from UMN [2] dataset where different behaviors (i.e., normal and abnormal) of the crowd are highlighted in *Frames 200-380*.

sion. Behavior understanding in crowded scenes includes the basic components, such as motion analysis, and recognition of behaviors (e.g., normal, abnormal, running, and dispersion) at the individual and collective level, is a problem that arises in a variety of different contexts and poses significant challenges to safety officials from the scene monitoring point of view. Infact, security and incident management for huge gatherings are a daunting task due to the dynamics of crowd formed by the large number of individuals as shown in Figure 1.3.

Many multi-disciplinary studies have been conducted to reduce the incidents of any catastrophic event in crowded situations through modeling the human psychology with the expected bottleneck areas [6]. However, such efforts are proven to be unsuccessful in numerous events of stampedes in the recent past which did not handle the management of crowds fully. For instance, in the year 2006 at Jamarat Bridge, Makkah, an event of stampede killed at least 346, and injured 289 [7], where two million people were performing the ritual at the same time. Similar kinds of stampede events are observed during social gathering around the world where the panic and lethal crush result in massive deaths and injuries. During last decades, many public places are constantly monitored by cameras due to security concerns which makes us believe that computer vision algorithms can contribute significantly towards the management of crowds and their behaviors, and can work in consonance with the safety officials. Moreover, crowd behavior analysis methods in computer vision research envision to automatically detect the events of anomaly that assist and help the safety officials to take quick actions.

The main motivating factor for developing the algorithms, specifically for crowded scenes is the limitations of existing surveillance systems. In general, state of the art surveillance systems have been developed to estimate the behaviors of objects and people in the scene, in isolation or in groups. However, when the video sequences are analyzed for crowded scenes, these systems are not appropriate and the performance is usually degraded as soon as the density of objects in the scene

increases [8]. Automated surveillance systems for crowded situations are almost non-existent when taking a quick glance at the research literature and industrial applications [6]. In recent years, quite limited research efforts have been spent in building computer vision systems to model the crowded scenes which provide useful information for safety officials. One obvious factor is the complexity and the inherent challenges due to the evolving dynamics of crowds which are the main obstacles in the direction of research efforts.

1.2 Challenges

Successful techniques for handling the non-crowded and crowded scenes must address a variety of problems, such as,

Foreground Detection: The extraction of focused entities is a crucial task in analyzing the scene for higher level vision tasks, such as tracking, action recognition, inferencing, and understanding behaviors in dynamic scenes. The first challenge is the foreground detection under diversified situations, for instance, gradual and sudden variations in light, small movements of non-static objects, abrupt or permanent variation in backgrounds. However, the criterion of performance is based on how robust an approach is under these situations.

Visual Features: The expedient representation of objects is an essential and challenging task in the course of developing robust tracking and behavior understanding framework for non-crowded scenes. Conflicted situations can result in the loss of correct estimation of visual parameters. However, if properly handled, more than one characteristic of the objects can be combined for the improvement of matching outcomes during tracking and behavior understanding of objects.

Ambiguity and Uncertainty on Object Matching: Ambiguity and uncertainty are raised from partial knowledge due to insufficient information or hazy contents. During the conflicted situations, the incomplete visual information about the occluded object makes it difficult to find the correspondence with previous objects and results in incorrect matching outcomes. This entails us to elaborate the qualitative approaches to handle these situations when the visual information is incomplete.

Choice of Granularity: The crowd, particularly human formed crowd exhibits both self-evolving dynamics and psychological characteristics which are directed to the similar goal, often. In addition, crowded scene contains complex interactions and frequent occlusions among the objects which make the computed features, such as interest points, localized heads, and specific human classifiers unreliable. Therefore, it is very challenging to come up with an appropriate level of granularity for modeling the dynamics of a crowd. Should a pixel-based model, individual-based model or something in between be used? It is essential to answer these questions for appropriate modeling of crowds.

Representation of Crowd Behavior: In the crowded scenes, complex interactions among individuals are indiscernible, and therefore individual's centric analysis of behaviors in crowds is implausible. In addition, the behavior of the crowd (i.e., normal and abnormal) often extends quite randomly, which makes it even more challenging to develop a general definition of the specific behavior by gleaning the information from an individual behavior.

1.3 Overview of Our Approach

In this dissertation, we have developed the frameworks to understand the behaviors of objects in non-crowded and crowded scenes. Unlike the typical holistic approach [9] [10] to process a video sequence for surveillance, we start by performing the background modeling to extract the foreground objects in the scene. The non-holistic level analysis eliminates the need for generating a representation of the whole scene in an efficient manner. In particular, this is achieved by developing a segmentation algorithm that exploits the idea of weighted integration of segmentation approaches to extract objects under complex situations. The segmentation is then used by feature extraction approaches where color of the object is the key feature for non-crowded scenes, and motion is the fundamental feature for crowded scenes. There on, this information (i.e., foreground and features) is exploited by high level algorithms (i.e., the framework) enabling objects tracking and behavior understanding for non-crowded scenes, and localization of normal and abnormal behaviors for crowded scenes. The repertoire of algorithms is proposed by employing the concepts of human cognitive behaviors with statistical approaches for vision system with the application to surveillance and object behavior unders-

tanding. Finally, we have proposed algorithms for crowd behavior understanding where the observed flow field in spatio-temporal bocks is parameterized with mixture of Gaussians to constitute a sequence of flow patterns which are classified by Support Vector Machine (SVM) [11] and Conditional Random Field (CRF) [12] to characterize the crowd behaviors. The performances of the proposed frameworks are demonstrated on the benchmark datasets along with quantitative and qualitative analysis.

1.4 Research Contributions

In the following, the contributions introduced in this dissertation are presented.

1.4.1 Tracking and Object Behavior Understanding

The framework developed in the context of tracking and behavior understanding for non-crowded scenes makes the following contributions:

Integrated Quantitative and Qualitative Approaches: The important contribution is the integrated quantitative and qualitative approaches which are motivated to treat the tracking problem by axiomatizing, and reasoning the human cognitive abilities. In quantitative approach, the matching weights are computed using proposed Bayesian Matching Weight (BMW) method. In the qualitative approach, the tracking axioms are proposed to handle the vagueness in the object matching when visual contexts of objects are lost, explicitly. Essentially, this feedback refines and handles the conflicted situations to disambiguate the object behavioral states during tracking. Both, the quantitative and qualitative algorithms are bi-directionally linked and complement their functionalities dramatically.

Bayesian Matching Weight (BMW): The proposed BMW approach employs Bayesian inference to measure the posteriori probability of objects and is referred as matching weights of objects. We have used two different approaches within Bayesian inference: 1) likelihood is computed as Kullback-Leibler (KL) Divergence [13] among the objects detected in consecutive time instances, and 2) prior information is measured with the Color Structure Code (CSC) approach [4]. This methodology works efficiently in the situations where the object proximity and scale changes, considerably.

Formulating Tracking Axioms in Qualitative Approach: In the qualitative approach, human perception, learning, and knowledge acquisition abilities for inferencing entities and its contexts are formulated into tracking axioms. These axioms influence the object matching ambiguity effectively by assigning the correct object's identities, and their respective behavioral states. The proposed tracking axioms are developed by taking into account the human cognitive abilities which provides an efficient mechanism to handle the vagueness in quantitative approach.

Kalman Filter-Based Tracking System: Objects are tracked by employing Kalman filter in our problem, but it suffers from the violation of linearity conditions. In object tracking, each detected object with unique identity is modeled as a linear system and Kalman filter is used to estimate the state of the objects at each time instance. In this manner, the proposed approach maintains the linearity condition even during the non-linear situations. Thus, we have extended the applicability of classical Kalman filter for both the linear and non-linear system without making any modification in its actual content.

1.4.2 Crowd Behavior Understanding

The framework developed in the context of crowd behavior understanding makes the following contributions:

Flow Computation with Uncertainty Handling using Social Entropy: The idea is motivated by observing the motion perception phenomenon under the influence of time dependent optical flow that reveals qualitatively, the dominant dynamics in spatio-temporal space. So, the motion is computed using optical flow approach [5] at the frame level (i.e., globally) over the detected foreground. Moreover, at the heart of our approach, we apply Social Entropy (SE) [14] to address the issues of optical flow noise. The concept of SE is originated from the field of social sciences which we have used to empirically determine a quantitative metric and it allows us to handle optical flow noise.

Modeling Flow with Mixture of Gaussians: The next contribution is related to optical flow modeling for computing the flow patterns. The mixture of Gaussians is employed to uncover the organization of the flow cloud data in each

spatio-temporal region named as the flow-block. The mixture of Gaussians encodes the flow cloud data into flow patterns in a manner that helps in revealing the representative characteristics of crowd dynamics.

Behavior Classification: The last contribution is in the context of characterization of crowd behaviors. We have treated crowd behavior understanding and anomaly detection as two-class problem. Both Hidden Markov Model (HMM) and Linear Discriminant Analysis (LDA) need strict requirements of conditional independence among the observed flow patterns. So, we have modeled the crowd behaviors using SVM [11] and CRF [12] to localize the crowd behaviors. Moreover, the performances of both classifiers are compared quantitatively and qualitatively.

1.4.3 Indirect Contributions

In this dissertation, frameworks are developed by adopting a top to down approach, and started by performing a low level analysis[1]. Moreover, segmentation and feature extraction are essential components which are later input to behavior understanding frameworks. So, state of the art approaches are investigated for foreground detection and weighted integration approach is proposed that combines the approximated median filter approach with adaptive background mixture model approach at the detection level. In this way, the segmentation is unaffected by the problems of erroneous detection of static pixels and ghost regions in the image when foreground object starts moving after a long period of time. Finally, an analysis and comparison of suggested approach are performed with related approaches.

1.5 Outline of the Dissertation

This chapter introduced the behavior understanding problem for non-crowded and crowded scenes, and described the key motivations and overall goals of this dissertation. The outline is structured as:

Chapter 2: In this chapter, state of the art approaches are categorized according to the adapted research strategy. We have divided this chapter into two main

[1] By low or local level analysis, we mean the non-holistic approach that begins with segmentation and feature extraction modules. These two modules are linked to high level analysis, which include the matching, recognition, classification, and inference modules.

levels: 1) literature survey and discussion, and 2) related issues. In the first level, we have reviewed the related literatures for segmentation, feature selection, behavior understanding and tracking for non-crowded scenes, and crowd behavior understanding and anomaly detection. Moreover, we are aspired to categorize the reviewed literature based on the methodologies used to develop the solutions, provided a detail description of representative methods in each category, and examined their pros and cons. In the second level, the related issues are underlined based on the analysis and narrates our research objectives more precisely.

Chapter 3: In non-holistic scene, object extraction from the background is an essential requirement to perform object tracking and behavior understanding. In this chapter, we have proposed a segmentation approach for both non-crowded and crowded scenes.

Chapter 4: In this chapter, the visual features employed for non-crowded and crowded scenes are described. Based on our analysis in Chapter 2, we have presented the fundamental concept of feature fusion in Bayesian inference approach for non-crowded scenes. Moreover, we have presented the idea of modeling the flow field by mixture of Gaussians to obtain flow pattern as feature for crowded scenes.

Chapter 5: In this chapter, we introduce algorithms in a unified framework that are specifically designed for tracking the individuals and to understand their behaviors in non-crowded scenes. In addition, this chapter discusses the steps involved in the construction of tracking axioms, and show how they can be integrated into a typical tracking methodology. Results are demonstrated on the challenging datasets gathered from state of the art resources along with the performance evaluation.

Chapter 6: This chapter presents the proposed framework for crowd behavior understanding, discusses the assumptions, and details about the steps involved in the mathematical modeling of crowded scenes. The results are presented on challenging benchmark datasets along with the comparative analysis and evaluation.

Chapter 7: In this chapter, we have concluded this dissertation with a summary of contributions and the description of future work.

CHAPTER 2

State of the Art

Our contributions to behavior understanding in non-crowded and crowded scenes center on the efficient tracking and behavior understanding of video sequences. The proposed framework is based on a top to down scheme that begins with foreground detection and visual feature extraction which are linked to high level behavior analysis. This dissertation is focused on various aspects that are crucial to perform object tracking and behavior understanding. We have provided an in-depth study of different segmentation and visual features computation approaches, tracking and behavior understanding for non-crowded and crowded scenes. Our objective is to provide a basic understanding for the following four chapters. Moreover, a detailed technical presentation of related issues are provided which are the motivation for the proposed framework.

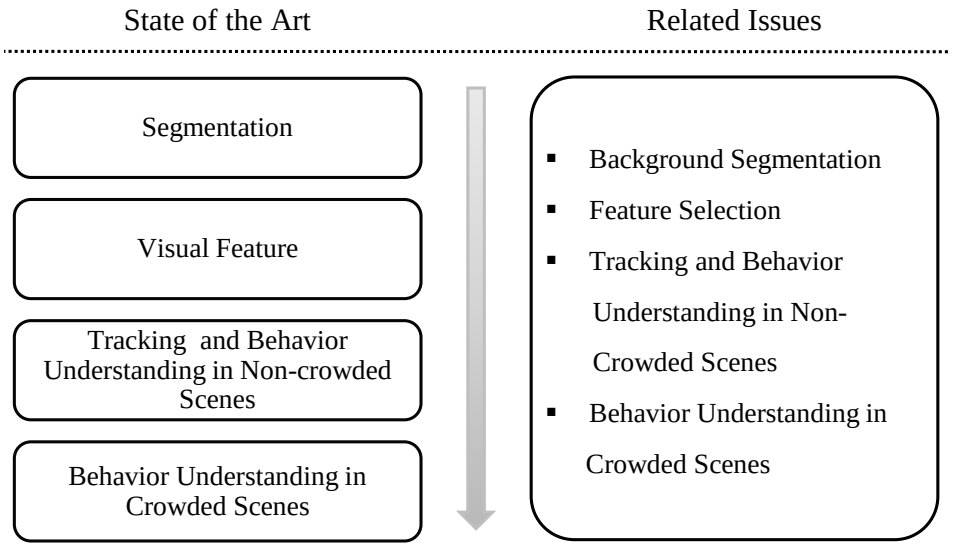


Figure 2.1: presents the organization of reviewed literature.

We have organized this chapter according to the development strategy used in this dissertation as shown in Figure 2.1. Section 2.1 gives an overview of segmentation approaches using background modeling and group them into appropriate categories. Section 2.2 presents the related literature on feature extraction for non-crowded scenes. Section 2.3 presents the related literature on feature extraction

for crowded scenes. Section 2.4 reviews the behavior understanding through object tracking for the non-crowded scenes. Section 2.5 presents the methodologies suggested for behavior understanding in the crowded scenes. During the literature survey, our key findings (i.e., related issues) are presented in Section 2.6 that define our motivations and contributions with practical perspectives. Section 2.7 ends this chapter with conclusion and discussion.

2.1 Segmentation Using Background Modeling

Much literature is available on segmentation [15] but we keep our focus on segmentation approaches using background modeling and provide an analysis of these approaches for non-crowded and crowded scenes. In video sequences, it is assumed that the background is monotonous and the foreground is exhilarating [16] as shown in Figure 2.2. So, the segmentation of foreground is obtained by taking a difference between the image of video sequence and the estimated background using an opportune thresholding procedure.

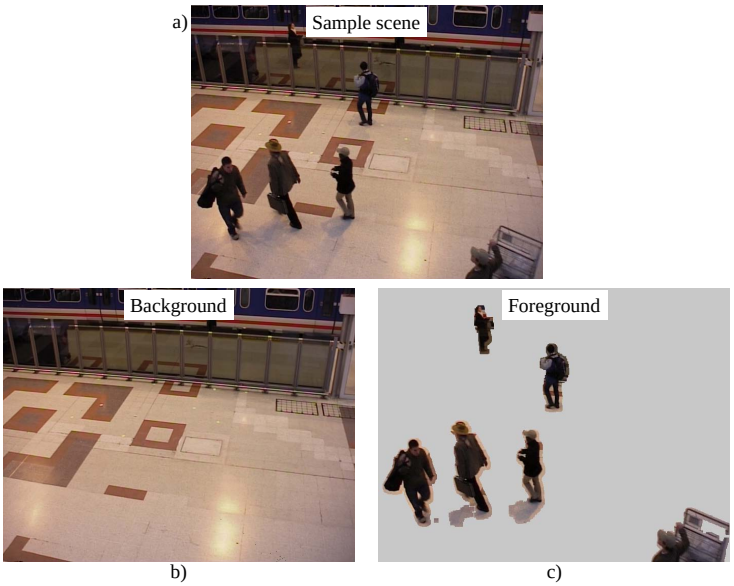


Figure 2.2: presents an example of scene components on a sample frame of PETS2006 [1]. a) shows the actual scene, b) indicates the background containing static part of the scene, and c) shows the foreground containing moving components. It is observable that some of the background components are also included in the foreground due to moving shadows and misclassified foreground pixels.

The classical background subtraction technique allows the extraction of fore-

ground region (i.e. object of interests[1]) from a scene. For example, the simplest estimate of background could be just an image of the scene without the foreground region. Fundamentally, pixel's values in each image of a video is subtracted from an estimated background model. Mathematically, the foreground is yield as follows:

$$|I(k) - B_{model}| > V_{th}; \tag{2.1}$$

where $I(k)$ and B_{model} are the image and estimated background model respectively, and V_{th} is an empirical threshold.

In many vision tasks, subtracting an estimated background from the image may result in good segmentation. But, the question is how to estimate or model the background? And is it enough to keep estimated background static? The static background subtraction leads to poor results when the underlying scene's background is not stable over time [18] due to light variations, camera trembling and outdoor conditions of the scene. In the literature [19], a variety of methods have been suggested to address these generic issues in real visual scenes. However, the criterion of performance is based on how robust an approach is under such situations. In the following sections, we have focused on the related state of the art approaches while these methods mainly differ in their background model type and updation procedures.

2.1.1 Non-recursive Techniques

In non-recursive techniques, a sliding window of any arbitrary size is utilized to estimate the background based on the temporal variation of each pixel within the window. Moreover, the size of the window should be significant enough to handle the slow moving objects in the scene. A series of research work is done in this direction which employs median filtering for background modeling. For instance, Cucchiara et al. [20] proposed an approach in which the background is estimated by taking the median at each pixel from all the frames in certain time window with an assumption that the pixel remains as background for more than half of the frames in the window. Figure 2.3(a) shows the result of this approach [20] over the sample scenes.

[1]The result of segmentation is the foreground region. In literature [17], the terms blob and objects have been used frequently. Some researchers argument that a blob or region can contain multiple objects in it. But, in this dissertation, we assume that the detected foreground region is object whereas any non-object region is termed as segmentation error.

Figure 2.3: presents the segmentation outcome of non-recursive approaches on PETS2006 [1] and PETS2009 [3] datasets. a) shows segmentation result of median filtering approach [20] where the learning time delay caused a ghost object which is not presented in the actual scene. b) shows the segmentation result of the approach [21] where over segmentation is observed along with additional noise in the final foreground detection.

With a different motivation, Elgammal et al. [21] proposed a holistic approach to form a non-parametric estimate of pixel density by utilizing the entire history of images in a sequence. The pixels are declared as foreground if they fall in the distribution smaller than some predefined threshold. The median of the absolute differences between successive frames is used as the width of the kernel in this distribution. Moreover, the computational complexity of modeling the background is similar to median filtering whereas the detection of foreground pixel is more complex as it is computed for each pixel. Figure 2.3(b) shows the result of this approach [21] over the sample scene.

2.1.2 Recursive Techniques

In recursive techniques, a single background model is updated based on each input frame or a window of frames for the segmentation. The initial research efforts [22] [23] have tried to address the background modeling and updation by utilizing recursive methods. For instance, McFarlane and Schofeld [22], proposed a simple recursive filter to estimate the median for modeling background and detection. Later, Remagnino et al. [23] modeled the background to segment urban traffic monitoring scenes in which 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. The resulting estimate is a converged value (i.e., median) for which half of the input pixels are larger than this value, and half of them are smaller than this

Figure 2.4: presents the results of recursive techniques on PETS2006 [1] and PETS2009 [3] datasets. a) shows the segmentation result of approximated median filtering approach [22] where an object is fragmented into parts due to slow pace. b) shows the segmentation results of the approach [26] where under segmentation (i.e., empty regions of holes in objects) is observed due to the constant region in the object.

value. Figure 2.4(a) shows the implementation of approach [23] over the sample scene. In a slightly different flavor, various authors used Kalman filter to model and automatically update the background for dynamic streams. These approaches mainly vary in their state space model, for instance, Karmann and von Brandt [24] used intensity and its temporal derivative to model the background. In contrast, Koller et al. [25] employed adaptive intensity and its spatial derivatives to model background according to the known parameters of weather, such as day light and darkness.

Moreover, there is another interesting and relevant body of work employing multi-modal statistical techniques to model the background. These approaches describe the consistent behavior of per-pixel background, and the foreground is marked when the pixel value does not belong to any background distribution. A classical work named as Running Gaussian Average (RGA)is presented by Wren et al. [27] which maintains a multi-class statistical model for the tracked objects where background model is represented by a single Gaussian. Since then, the idea of employing Gaussian model [28] for background has gained tremendous attention of researchers. For instance, Friedman and Russell [29] modeled the pixel intensity as a weighted mixture of three Gaussian distributions (i.e., each for road, shadow, and vehicle) for traffic surveillance. Similarly, Stauffer and Grimson [26] created a multi-modal background using mixture of Gaussians. Each pixel is compared with every Gaussian in the background model until a matched Gaussian is found.

Mean and variance of the matched Gaussian are updated otherwise a new Gaussian is created with initial mean (i.e., current pixel color) and initial variance (i.e., empirically selected). The pixels are classified as foreground, if they do not match with the distribution in the background model. Figure 2.4(b) shows the implementation results of this approach [26] over the sample scene. A wide range of research [30] [31] [32] has been done to modify and improve the classical concept of mixture of Gaussians-based background modeling. The performance of these approaches is superior than traditional difference algorithms, thus making the problem of threshold selection less critical. However, good empirical parameters are pertinent to initialize these adaptive mixture models for the optimized learning and cohesive foreground segmentation. To address these limitations, our proposed approach combines two approaches to perform segmentation in Section 3.1.

Discussion

Most of the high level tasks in vision used fixed camera [33] where the extraction of foreground is an essential task. To start with, non-recursive methodologies [20] [21] have been used and explored for segmentation. In practice, these approaches require continuous object motion in successive frames. However, when the objects remain static, incomplete object regions are observed in many instances, such as objects may be fragmented into several regions, or there can be ghost regions in the image. As a result, the detected object may include the background in it, significantly, and there will be no guarantee that the detected region is the foreground as shown in Figure 2.3. In contrast, recursive approaches are widely used to model the background and to extract the foreground. In earlier attempts [23] [27], single Gaussian is used to track the evolution of background. However, multiple colors may be observed at a certain location due to repetitive object motion, shadows or reflectance. Therefore, a single Gaussian is not sufficient to model the background [34] for outdoor scenes. To address these issues, mixture of Gaussians is used in [35] [36] [32] and are further extended by [30] [37]. Indeed, mixture of Gaussians results in effective segmentation of foreground for the dynamic scenes which account for both multi-modality and clutter situations. Moreover, the recent approaches [30] [37] are able to model the background under changing illumination, noise, and periodic motion of the background regions. However, these approaches are sensitive to the corresponding known parameters to initiate and update the background model.

2.2 Visual Features for Non-Crowded Scenes

In tracking and behavior understanding, unique features play a significant role. Because, the uniqueness of a feature under diverse situation ensures that the object can be distinguished over time. From last two decades, color, shapes, motion, and texture are the commonly used features. Among these features, color can be considered as a simple and effective representation of an object. Moreover, we have described some practical approaches for detecting and modeling these features to ensure efficient feature correspondences over time. In the following, we have discussed in detail the relevant features along with the fundamental concepts.

2.2.1 Color Modeling

The color of an object forms a continuous visible spectrum that is influenced by the physics of the object, its environment, and the characteristics of the perceiving eye and brain. In the digital world, pure spectral colors are mapped into a predefined color spectrum, namely color space which is divided into distinct color values, for instance, RGB (Red, Green, Blue), HSV (Hue, Saturation, Value), and YIQ (Y-axis, In-phase, Quadrature).

We have used HSV color space, inspired by the way human visual system perceives the object. In HSV, Hue is a color attribute which represents the dominant color. The saturation is an expression of the relative purity or the degree to which a pure color is diluted by white light. The brightness is a measure of luminous intensity and embodies the achromatic notion of intensity. In the HSV color space, the brightness component is separated from color-carrying information (i.e., Hue and Saturation). In Section 4.1.1, we have demonstrated a comparative analysis among color spaces (i.e., gray scale, RGB, and HSV) for similarity measure and it is found that HSV color space is better in performance. In the following, we have explained the color modeling approaches along with their categorization.

Color modeling gives an effective representation of an object's color characteristics and is preferred over an individual feature[2]. The common approaches for color modeling are:

- Histogram: This approach is used to nonlinearly map the probability distribution of the color spectrum of an image or a pattern. In general, the effect of the histogram is homogeneous as all the pixels are subjected to the same

[2]It is not feasible to take into account the color values of each pixel of any object.

functional mapping resulting in a compact representation of image characteristics without requiring knowledge about them. Histogram is widely used in many applications, for instance, in object tracking [33].

- Parametric Modeling: In this modeling, a specific functional form is assumed to model the probability density. A model is defined by several unknown parameters, and the given data is used to compute these unknown parameters. However, it is in general difficult to find a suitable model and its parameters if the distribution of data is not given. One of the most commonly used parametric modeling approaches is mixture of Gaussians, whose parameters (i.e., mean and standard deviation) are optimized with Expectation Maximization (EM) approach [38].

- Non-parametric Modeling: Kernel density estimation is a non-parametric technique that does not assume any specific model. Kernel density estimation is used in tracking, for example, Comaniciu et al. [39] proposed an approach in which kernel density is computed by summing the probabilities computed from individually modeled observations.

- Color Structure Code (CSC) Approach: In this approach, the object is segmented into regions based on its color distributions [4] where homogeneity or gradients have to be defined on vector valued functions for color images. CSC approach has been used as the object descriptor in tracking problems [17].

The color based modeling approaches described so far result in a model which is considered as a single entity inspite of taking many individual components (e.g., the pixels color) into account. Among different approaches, in the context of tracking and behavior understanding of objects in the non-crowded scene, we have employed the color histogram [40] and CSC [4] approach due to their flexibility and robustness. In Section 4.1.1, we have described the formulation of color histogram and in Section 4.1.2, we have explained CSC approach. Both of these features are exploited in our tracking and behavior understanding algorithms.

2.2.2 Geometrical Features

Finding the geometrical features, such as contour, area, and boundaries of the detected objects in images are apparently simple task but noise and poor segmentation degrades the efficiency which results in errors. Inspite of the apparent diversity of

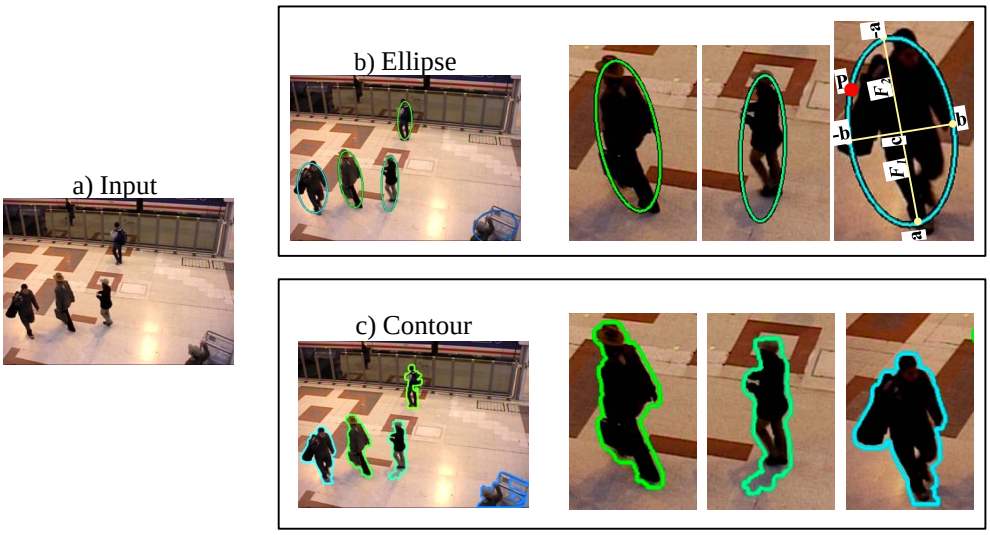


Figure 2.5: shows the visual mapping of computed features (i.e., ellipse and contour) on a sample frame of PETS2006 [1]. a) is the input image, b) demonstrates the computed ellipse over the detected object given the contour of object as input data. We have also labeled the parameters of the ellipse on object and c) presents the results of computed contour around the detected object.

geometric features, it turns out that some basic features are used to support[3] high level tasks, such as tracking, behavior understanding, and crowd analysis. Among many geometrical features, we have explained the following features used in this research[4].

- Ellipse: Ellipse is a smooth closed curve resulting from the intersection of a circular cone by a plane which is symmetric about its horizontal and vertical axes [41]. The distance between any pairs of points whose midpoint lies at the center of the ellipse, is the maximum along the major axis and minimum along the minor axis. Mathematically, on the ellipse, two special points (i.e., F_1 and F_2) are defined on the major axis called foci. These points are equidistant from the ellipse center (i.e., c) and separated by a distance. Given a point P on the ellipse called a focus, the sum of the distance to these two foci is constant and equal to major diameter ($d_1 + d_2 = 2a$), where a is major axis whose origin is at one of the foci and b is semi minor axis. Figure 2.5(b) shows computed ellipse around the detected objects.

[3]The geometric features, such as bounding box can be employed to define the search space criterion.

[4]The features are computed on the segmented outcome obtained from weighted integration segmentation approach in Section 3.1

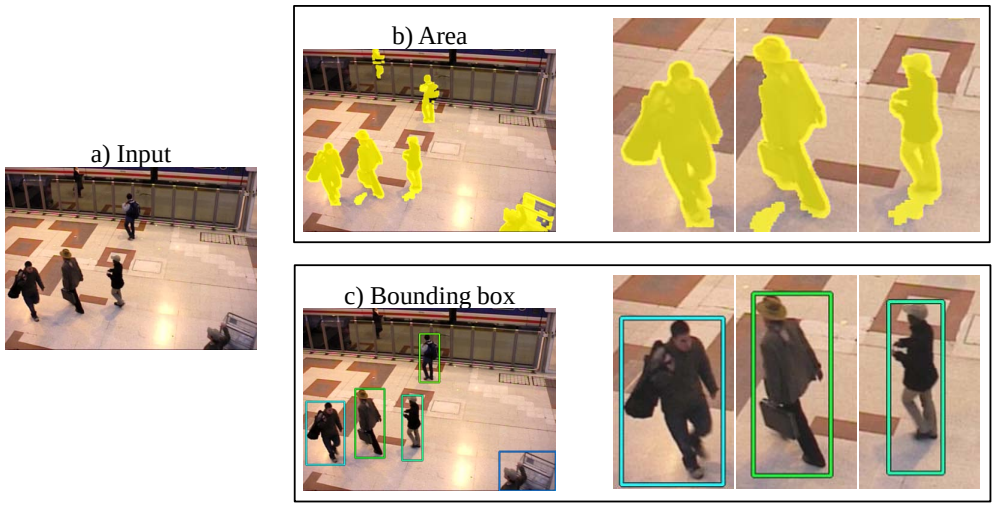


Figure 2.6: shows the visual mapping of computed features (i.e., area and bounding box) on sample frame of PETS2006 [1] sequence. a) is the input image, b) demonstrates the computed area (i.e., transparent yellow color) of the corresponding detected objects, and c) presents the results of computed bounding region around the detected object. It is notable that due to strong shadows the object's area and bounding region includes some region of background as well (in the right most zoomed object).

- Contour: Contour of a binarized segmented object represents a close boundary measured using the chain code technique [41]. Generally, chain codes are represented based on 4-or 8-connectivity of the segments (i.e., neighbors) where the direction is coded by using a numbering scheme. Moreover, it follows the boundary of object in clockwise manner and assigns a direction to the segment that connects every pair of pixels resulting in the contour of object. Figure 2.5(c) shows computed contour around detected objects.

- Area and Bounding Region: Area of the object is calculated from the objects pixels as shown in Figure 2.6(b). It is invariant of translation and rotation [41]. The bounding box of an image represents the fully enclosed rectangle which includes the detected regions in the image in Figure 2.6(c).

2.3 Features for Crowded Scenes

Many authors have proposed the methods for detecting the crowd behaviors by applying state of the art feature set [42]. For example, detection of interest points,

Figure 2.7: shows an example that highlights some of the challenges in the crowded scenes on a sample frame of PETS2009 [3] dataset. a) and c) indicate the complex occlusion. b) and d) show the interactions among individuals in the group.

detection of the heads or legs, and color histogram. However, all of these features tend to be impractical under crowded scenes due to the complex interactions of objects as shown in Figure 2.7.

Therefore, researchers are more interested to compute and model the optical flow at various levels (i.e., local, global and spatio temporal cubiods) to overcome these shortcomings. However, computing an efficient and fast optical flow is crucial. Besides, we contend that the behavior description at the individual level may not be necessary, and thus, modeling the crowd at the spatio-temporal level is more practical while the holistic approaches are computationally expensive. We have computed the optical flow on the detected foreground and modeled flow cloud data which acts as the fundamental feature for crowd behavior understanding (i.e., normal and abnormal behaviors) as described in Section 6.2.

2.3.1 Motion Modeling

Motion of objects in a video sequence is the source of temporal variations in a scene. Typically, we are interested in determining the relative motion between camera and objects in the scene which are computed efficiently if the corresponding parameters, such as rotation and translation are known. In contrast, the extraction of motion comes down by projecting the actual motion onto 2D image plane. This projective representation of actual motion describes the apparent motion or commonly referred as optical flow. Comprehensively, optical flow is specified by observing the pixel motion between two adjacent images or simply how much change is observed in image intensities.

There are numerous computational models that have been developed in the literature to estimate the motion from a video sequence, such as local and global differential methods [43] [44], feature-based techniques [45], layers-based ap-

proaches [46], and phase-based approaches [47]. Differential methods provide the benefit of transparent modeling and rotation invariance as well as better qualitative performance that makes the superiority of differential methods succinct [48].

Particularly in differential methods, accurate and consistent estimation of motion is challenging due to many factors, such as motion discontinuities, aperture problems, and large illumination variations. These factors lead to direct violation of assumptions which are taken into account by the prototypical approaches, such as Horn and Schunck [43] and Lucas and Kanade [44] techniques. Therefore, a variety of models have been proposed for effective flow computation, including robust functions [49], integrating gradient information [50], estimating a symmetric flow field [51], combining local and global flow [52], and reasoning about the pixels under the unusual situation, such as occlusion mode [53]. On the available sequence (i.e., Yosemite sequence), the successful computational measures are presented with an average angular error of 2° [54]. However, the situations and conditions are not discussed where these algorithms fail. In the following, we have explained briefly the commonly used flow modeling approach:

- Flow Histogram: The idea behind flow histogram is motivated by the typical color histogram. After performing the flow field computation, both flow magnitude and direction of motion are quantized using 2D optical flow histograms named as Histogram of Flow (HOF) descriptors. Many researchers have used the flow histogram for the non-crowded and crowded scenes to determine the object specific activities, such as walk, run, fall, and unusual actions. For instance, HOF is used to match the motion of a player in a soccer match [55] whereas [56] classifies the histogram of oriented flow for human action recognition.

- Mixture of Gaussians: Parametric modeling, particularly mixture model is one of the commonly used approaches where many methods are proposed to optimize the performance of the modeling process. For instance, the parameters of mixtures are initialized by applying the clustering techniques such as K-means whereas the optimization is performed the EM algorithm. The main objective of representing the flow field in mixture of Gaussians is to provide a vivid representation of huge and replicating flow field. Alternative to the mixture of Gaussians, Principal Component Analysis (PCA) is also used for the similar purpose. However, the computation cost of estimating mixture model parameters is usually more expensive. In addition, the number

of components used to represent the data is hard to estimate. Recently, in this work, Saleemi et al. [57] have exploited mixture of Gaussians to represent the dense flow field for unusual behavior recognition in the non-crowded scenes.

- Computational Flow Model: In sociology and behavioral sciences, modeling of observed dynamics particularly in crowds is quite researched and an active topic. Over years, a number of models have been developed for various situations with applicability, such as crowd management in Makkah or more recently marriage ceremony of Prince William. For modeling the crowd, discrete simulation of individual objects is an established methodology. A popular method of modeling flow field is Cellular Automata (CA) where the local movements of the objects are modeled with a matrix of preferences, for instance, the probabilities for related walking directions, speed, and goal-oriented directions. Another famous model is the Social Force Model (SFM) in which the individual is under the effect of long-ranged forces, and its dynamics follow the equation of motion, similar to Newtonian mechanics. Various researchers have exploited this concept, such as Ali et al. [58] proposed a method for crowd segmentation, and Mehran et al. [59] detected the anomaly in crowds.

The flow modeling approaches described so far result in a model which is considered as a concrete representation of flow field. Among different approaches, in the context of crowd behavior understanding, we have employed the concept of mixture of Gaussians to marginalize the flow cloud data in Section 4.2.

2.4 Tracking and Behavior Understanding in Non-Crowded Scenes

Behavior understanding in the non-crowded visual scenes is one of the important research domains of computer vision where tracking is a primary element. Briefly speaking, tracking of object aims to generate an inference about object's motion in corresponding video sequences. However, practically, it is a difficult problem due to the interferences during object movement in the scene such as occlusion which is observed among objects very frequently as shown in Figure 2.8. There are various types of occlusions such as: 1) object-to-object occlusion where objects intercept their appearances during motion, and 2) object-to-scene occlusion

a) b)

Figure 2.8: shows various examples of direct interference (i.e., pointed by arrows) on the sample frames of PETS2006 [1] and PETS2009 [3] datasets. a) indicates the motion variation of an object and occlusion among the objects at the same time; b) indicates the situation where the orientation of the object is changed significantly.

where object appearance is occluded by the corresponding obstacles in the scene such as trees, walls, stands and other stationary objects. To address the later type of occlusion, modeling the scene geography is an essential requirement [60]. However, in this research, we have focused on the object-to-object occlusion problem during tracking; therefore, the topic of scene modeling is out of the scope.

The tracking algorithms usually follow a modular scheme which either perform Detection prior to Tracking (DpT) or Tracking prior to Detection (TpD) in a flexible architecture. In the DpT approach, the objects of interest are first detected at every instance of time and tracking of the detected objects is performed, only. In contrast, in TpD approach, a hypothesis is built about the object location in the generated state space which is then evaluated by a computed set of features in an image. Moreover, in recent years, object identity management is also gaining attention by incorporating the concepts of data association through similarity measurements. The objective is to assign unique identities to each object and to manage these identities over time. Later, elementary trackers (i.e., Kalman filter or Mean shift filter) are used to track these objects.

In the following, we have categorized the related tracking approaches in two main directions. First, recent developments in tracking algorithms based on DpT and identity management approaches are briefed along with their limitations in terms of generality, and complexity. Second, the approaches using logical models in computer vision, particularly for object tracking are examined.

2.4.1 Tracking and Identity Management Approaches

In this section, we present the fundamentals and review the DpT and object identity management approaches. In DpT approach, two processes are involved for multi-object tracking: 1) data association, and 2) object state estimation. The data association process addresses the correspondence problem by determining the informative measurement and deals with data uncertainty in the presence of noise and motion perturbation. The estimation of object's state deals with the measurement inaccuracy. Usually, the measurements consist of the object proximal attributes (i.e., location, speed, acceleration, etc.) in the image which is computed in pre-processing (i.e., detection and feature computation) levels. For instance, what is the true state of the object at a specific time instance based on the assumption that the object's measurement is correct?

Data association is probably one of the complicated problems in conflicted situations, such as occlusions, splits, enter or exit objects, and mis-detections. In such situations, the difficulties are raised because it is not certain that the measurement values convey information about the state of the object being tracked. In multi-object tracking scenario when elementary trackers, such as Kalman filter, Mean shift filter, or Particle filters are employed, it is essential to deterministically associate the most likely measurements for a particular object to that object's state. So, the data association problem needs to be solved before applying these filters. An incorrectly associated measurement can cause the filter to fail in the convergence. To address these issues, there exist several statistical data association techniques to tackle this problem. For instance, a detail review of these techniques is provided in Bar-Shalom and Fortmann [9]. Moreover, in the earlier attempts, Joint Probability Data Association (JPDA) and Multiple Hypothesis Tracking (MHT) are two widely used techniques for data association.

JPDA is a famous approach for multi-object tracking and is based on the Bayesian estimate to find the correspondence between the detected features where many targets are to be tracked. JPDA functions on some assumptions among many, such as: 1) the number of established targets are known in the clutter, 2) measurements from one target can fall in the validation region of a neighboring target where this situation can happen over several sampling times, and acts as a persistent interference, and 3) the states are assumed to be Gaussian with means and covariances according to the approximated system. These assumptions limits the applicability of JPDA approach in various real-scenarios.

In MHT algorithm, several correspondences for every object at any time instance are maintained [10]. These correspondences are established using only two consecutive frames which result in the finite chances of incorrect correspondences. For accurate tracking, the correspondence decision is deferred until several frames to examine the final correspondence [61]. As a result, the final track of the object is the most likely set of correspondences over the time period of its observation. Moreover, the algorithm has the ability to create new tracks for objects entering the field of view and terminate the tracks when objects exit the scene. It can also handle occlusions (i.e., continuation of a track) even if measurements are missing.

A wide range of literature has been published to handle the fundamental limitations of data association approaches [62]. For instance, Cox and Hingorani [63] have presented an efficient variant of MHT approach in which the k-best hypotheses are determined in polynomial time using Murty's approach. However, the suggested approach has limited the total number of hypotheses using pruning. With the similar motivation, Isard and MacCormick [64] proposed Bayesian multiple block tracking system. In their approach, a multi-blob likelihood function assigns the direct comparable likelihoods to hypotheses containing different number of objects. This likelihood function is adapted from the theory of Bayesian correlation. After that, Bayesian filter is used for tracking multiple objects when the number of objects is unknown and varies over time. This approach has some fundamental limitations, such as too many parameters assignment which play a crucial role in performance. Similarly, Smith et al. [65] proposed a Bayesian framework for the fully automatic tracking of variable number of interacting targets. They have employed a joint multi-object state-space formulation and a trans-dimensional Markov Chain Monte Carlo particle filter is used to recursively estimate the multi-object configuration and efficiently search the state-space. The demonstrated results are impressive, but the approach assumes that the motion is associated to one or more persons. However, the actual blobs may contain multiple categories of objects, such as shadows, reflection regions, and blobs due to camera motion parallax. More recently, Ryoo and Aggarwal [66] presented a paradigm for tracking objects under severe occlusion named as observe-and-explain. This approach has enumerated multiple possibilities of tracking by generating several likely explanations after concatenating a sufficient amount of observations. Further, the system chooses the hypothesis path with the highest probability which enables the tracking of even fully occluded objects with the cost of higher computational effort.

A different way to address the object tracking issues is through object identity

recognition. In the literature, not much work has been reported for object identity recognition in which a specific individual detected at certain time instance is matched with the previous observations. A framework is presented by Guo et al. [67] for vehicle matching in aerial views but their main focus was blob extraction and alignment rather than recognition. Gheissari et al. [68] presented a two layer method for human identification. In the first layer, a graph based spatio-temporal segmentation is applied to group the human pixels that belong to the similar cloth. The second layer used the decomposable triangulated graphs to segment and link different parts of the human body. Even though, human recognition is not the direct focus of the literature, but some seminal advances in human detection have been reported that can be indirectly associated. For instance, Dalal and Triggs [69] trained a SVM classifier using features of Histograms of Oriented Gradients (HOG) for human detection and localization. However, these methods are highly dependent on image details for extracting the features, such as faces or body parts, and therefore, can only be applied to high-quality images. Both, tracking and object identity recognition are closely related problems, since solving the tracking implicitly accomplishes the task of identity recognition and solving identification over consecutive frames is actually one of the fundamental tasks of object tracking.

Discussion

In the above, we have confined the underlying analysis to DpT and identity management approaches. In typical DpT approaches, the major limitations are in data association and estimation approaches under the cluttered, occlusion, or when the numbers of objects are higher. For instance, the major limitation of JPDA algorithm is its inability to handle new objects entering the field of view or already tracked objects exiting the view. Since JPDA, the algorithm performs the data association of a fixed number of objects tracked over two frames, serious errors can arise if there is a change in the number of objects. On the other hand, MHT is an iterative algorithm and makes the association in a deterministic sense which exhaustively enumerates all possible associations. Consequently, the MHT algorithm is computationally expensive both in memory and time.

Another limitation in data association techniques is that the similarity measurement criterion and estimation of object locations are solely based on the object features, which are usually referred as low-level features. These approaches function optimally in a flexible environment with a limited number of objects that are

being tracked, and search for these tracked objects within a spatial region. Moreover, such techniques have proven their efficiency in continuous scenes where disappearances and clutters are minor. However, under severe occlusions and in complex environments, these data association algorithms do not perform well and suffer from failure [70]. As, the problem shifts from solving the correspondences in a smooth continuous video to individual detected objects with long temporal gaps, all the assumptions of the continuous motion models become weak. So, the solution becomes the object identity recognition and management along with tracking approaches.

With different perspective, considering the object identity recognition problem in object tracking, we have to revise the theory of object identity recognition and infuse it with tracking mechanism implicitly. For instances, in tracking, objects are usually considered to have small displacements between observations which is not mandatory in object identity recognition. In contrast, object identity recognition methods are highly dependent on image details to extract features, such as faces or body parts, and therefore, it can only be applied to high quality ground images.

In the view of above mentioned analysis to handle data association (i.e., matching[5]) ambiguities during clutters and occlusion, we have introduced the concepts of human natural capabilities into our tracking framework. The idea is motivated by the fact that in real-world, humans with unbeatable natural capabilities utilize a number of so-called high level concepts while recognizing the objects around them. Logical modeling of human cognitive abilities is the discipline that focuses on developing methodologies and techniques to embed high level reasoning in association with the typical vision algorithm [9]. However, so far, such techniques are feasible and applicable only on context-based domains.

2.4.2 Tracking with Cognitive Modeling

Human beings always try to understand their environment, infer the actions and behaviors accordingly. Suppose viewing a street view from one's window, you are curious about which object enters in the scene and how a person disappears immediately while walking. This ability of object's movement understanding seems so natural and simple for ordinary people, but it actually requires complicated algo-

[5]The aim of data association is to find the matching among the observations through similarity measure where this term is widely used in the context of tracking. But, infact it is a methodology that can employ matching algorithms by exploiting the features of interest to find correspondences [71].

rithms in the field of computer vision and Artificial Intelligence (AI) to perform these tasks. Consequently, motivated by human perception, learning, and knowledge inferencing abilities, various researchers have modeled these human cognitive abilities and incorporate them into the vision research. The goal is to enable vision systems to have similar capabilities as humans for recognizing people's activities and behaviors. However, existing literature incorporating both vision and cognition model to address the problem of tracking is very limited.

Among many, computer vision was identified as the earlier goals of AI. The researchers make use of logical reasoning as a broad purpose mechanism for modeling intelligent behaviors to address the vision problem. In [72], a collection of papers is available for computer vision research involving the development of semantic representations and inference mechanisms. Specific to tracking, Haritaoglu et al. [73] proposed a real time surveillance framework for detecting, tracking, and monitoring people activities in an outdoor environment. Their approach takes the images from an infrared camera as input so, the color information is not taken into account unlike many systems for tracking people [33]. Alternatively, a combination of shape analysis is employed to track the people along with their body parts (head, hands, feet, and torso) and model of people appearance are created to ensure the tracking during interaction or occlusions.

Sherrah and Gong [74] proposed a framework for tracking the objects and handle the plausible interpretation of incomplete data due to the body parts interactions (i.e., hand and face) by enforcing explicit domain knowledge and high level semantics with Bayesian network. Thus, the efficiency of the suggested approach is claimed by activating the inferencing when occlusion is observed. However, the tracker assumes the whole high-dimensional state space to infer object positions which can be computationally inexpensive when detecting face and hands of an object. In addition, when tracking objects in the subways, the suggested approach can be inapplicable and expensive to compute due to complex manipulation.

More recently, Bennett et al. [75] proposed a technique by applying the principles of logical reasoning explicitly to rectify the imperfect output of an object tracker (i.e., far more accurate than the raw output from the tracker). Their suggested approach consists of three elements: object tracker, object classifier, and reasoning engine to handle ambiguities. Moreover, the ambiguity is defined when many objects merge due to occlusion and introduce the multiple hypothesis. So, their reasoning engine handles these ambiguities in a unified framework to produce a hypothesis by considering a globally consistent model that is maximally suppor-

ted by a voting function based on the output of a statistical classifier. However, in their work, uncertainties due to occlusion are disambiguated after classification through long-term reasoning unlike our proposed work which incorporates the logical modeling in parallel and more focused on exploiting the logics with the matching scores computed by employing the statistical model [76].

Discussion

The research in computer vision directly addresses the real-time issues that turned out to be much harder than the anticipated ones by the research community. It is observed that there has been a significant divergence between the fields of knowledge representation and computer vision. Unlike knowledge representation, computer vision research has moved away to the logical concept and employed statistical techniques for most of its algorithms. But, statistical methods alone have suffered with many apparent limitations as discussed earlier. Particularly, in handling the prediction of joint probabilities without considering the domain specific logical constraints can manifest errors and require intensive computation.

However, employing the logical framework under such situations (i.e., occlusion and cluttered) are relatively easy to state and reason with. Logical reasoning can provide a powerful mechanism for determining consistent possibilities. Moreover, the conceptual structure of possible situations is formulated and semantic knowledge is used to infer and guide for plausible interpretations (i.e., in data association) under occlusion situations. Our analysis at this stage is more biased towards logical approaches by considering efficacious performance even if the discrete data is incomplete and ambiguous. Thus, combining these two approaches complement each other which is the motivation behind the proposed research.

2.5 Behavior Understanding in Crowded Scenes

Despite of significant efforts done by vision researchers, the surveillance assumptions about the density of objects often violates in real-world scenes, for instance, Figure 2.9(a-b), shows example of crowded scenes where objects are moving in groups with some specific goal whereas Figure 2.9(c) indicates that many objects are gathered in unruly manner. Managing crowds and understanding their behaviors in public places are studied by many commodities, such as sociologists, psychologists, civil engineers, computer graphics and computer vision researchers for

a) b) c)

Figure 2.9: shows example of the crowded scenes containing objects in different contexts and situations.

many applications including visual surveillance [6]. However, maintaining security measures in these places is a daunting challenge to avoid catastrophic situations [77]. In the following section, we have investigated the approaches suggested for crowd behavior understanding and anomaly detection. We have categorized the research of behavior understanding in crowded scenes based on the methodology that each work has used to solve this task. These methodologies are described as follows:

2.5.1 Behavior Analysis with Individual Detection

These approaches are aimed to detect individual persons in the crowds and modeled their activities and behaviors. For instance, a model-based segmentation scheme is suggested to localize the individuals in crowded scenes by Zhao et al. [78] in a Bayesian framework. In their approach, a person is defined with their associated parameters maximizing the posterior probability which reflects the matching of 3D human shape models with the foreground blobs while preferring small number of objects. Their approach performs well on low dense crowds, but it is not scalable to high dense crowds where high inter-occlusion prevents the visibility of the complete human body quite often.

In previous years, it is observed that a number of researchers have experimented the behavior analysis of individuals in crowded scenes by detecting the interest points. For instance, a global annealing optimization framework is proposed by Tu et al. [79] using the clustering of interest points based on their (i.e., among) geometric associations to segment the individuals in crowds. In their approach, the crowd scene is taken from the top view. Therefore, such camera setup limits the applicability of the approach. In similar context, Brostow et al. [80] proposed an

unsupervised Bayesian clustering framework for grouping the trajectories of moving entities based on their space-proximity. In their approach, the image features are tracked and grouped probabilistically into clusters where space-proximity and coherence of the trajectory in image space is used as the probabilistic criteria for clustering. The results are reported on a low-density scene. Recently, Stalder et al. [81] proposed an adaptive grid-based classifier for object detection in crowds based on the local context. In their approach, different classifiers are trained incorporating various contexts over time, such as scene specific samples from the background, and object class. Moreover, these samples are used to update the specific object detectors and the results are shown on crowds having a normal walk.

Discussion

The main drawback of these methods is that they tend to be impractical in dense crowded scenes which contain objects with varying scales or interacting in the complex manner as shown in Figure 2.7. Because, most of these approaches are essentially designed to perform the detection of individuals in low dense crowded scenes. Moreover, the computed features, such as interest points, localized heads, and specific human classifiers become unreliable. To overcome this shortcoming, we argue that detection of individuals is not crucial; instead modeling the crowd at a global level is more practical in the dense crowd with complex interactions. For this purpose, we have proposed a framework which is capable of localizing crowd behaviors in a scene at the global and specific level.

2.5.2 Behavior Analysis with Trajectory Modeling

Over years, tracking algorithms are focused to perform surveillance on non-crowded scenes. In contrast, the particular challenges of surveillance in the crowded scene are not fully addressed. Some interesting works are reported that try to track the crowd of ants [82], flock of bats [83], and players in hockey ground [84]. However, most of these algorithms only use features (e.g., corners, contours, bounding regions, etc.) for tracking purposes.

Surprisingly, little work has been reported in exploiting high level cues for human detection, tracking, and behavior analysis in crowded situations. For instance, Antonini et al. [85] proposed the problem of detecting and tracking the crowds using discrete choice models for pedestrian behavioral patterns. In their approach,

prior knowledge of pedestrian dynamics is exploited to predict human motion patterns and integrate it with the visual tracker for robust performance. Later, Ali and Shah [58] suggested a methodology for tracking subjects by employing the strong assumptions on subject behaviors in high dense crowded scenes captured at distance. In their approach, a floor field is formed that captures the expected motion of the video subjects related to the physical nature of the scene. The results are reported on structured crowded scenes such as marathon. In contrast, Rodriguez et al. [86] proposed a framework for tracking unstructured crowded scenes in which various behaviors of crowds are mapped at different locations of the scene by employing Correlated Topic Model (CTM). In their approach, low level quantized motion features are treated as words and crowd behaviors determine the topics in CTM model. In this manner, each location in the scene supports multiple crowd behaviors and uses them as a prior information for tracking. The results are demonstrated on the footage of sporting events with strong restrictions (i.e., crowd should be coherent) on the behavior of the video subjects.

Recently, Wu et al. [87], proposed a framework for tracking and modeling the trajectories to localize the anomalies in crowds. Their method is constructed in three layers: 1) particles are advected based on the flow field, 2) the similar trajectories are grouped to obtain representative trajectory, and the chaotic dynamics are extracted and quantified with the maximal Lyapunov exponent and correlation dimensions, and 3) these chaotic features are learned by probabilistic model, and a maximum likelihood estimation criterion is adopted to classify the scene as normal or abnormal.

Discussion

In the crowded scenes, the objects are highly anticipated, and it is difficult to determine the low level features (i.e., color, spatial templates, interest points, contours, etc.) owned by the specific individuals keeping the reliability of features at risk. Moreover, due to the interaction among objects in the crowds, severe occlusions are observed frequently; therefore, tracking over longer time durations is difficult. To address these limitations, the authors use higher level knowledge and model the pedestrian behaviors into the tracking algorithm based on the strong assumptions about the pedestrian behaviors. In contrast, due to the high variability in pedestrian dynamics, crowded scenes tend to have less structure regardless of their similar density. The resulting track of individuals (i.e., trajectories) are highly in-

consistent and unable to discriminate between usual and unusual events. Therefore, we contend that the tracking-based models may disregard the important correlation between objects within close proximity of each other and is impractical to handle a wide range of situations under flexible assumptions.

2.5.3 Behavior Analysis with Modeling Crowd Flow

Several methods have been reported with alternative solutions to avoid the above discussed issues of detection and trajectory modeling in Section 2.5.1 and 2.5.2. The commonly used features in these solutions are optical flow, gradient, spatio-temporal volume to represent the dynamics of the crowd. In earlier attempts, Boghossian et al. [88] proposed a technique to model the dynamics of the scene by online-illustrations to pinpoint the crowd-related emergencies in large crowds. In their approach, the estimated optical flow is clustered based on direction and magnitude of the segmented crowd. Later, different events, such as circular flow paths close to site exits (i.e., trapped event crowds), crowd-flow diverging from a point to all directions (i.e., potential suspicious event, fights, fire), and obstacles in the flow paths (i.e., disturbance event) are detected by employing Hough voting. Using the optical flow, Andrade et al. [89] maintains a generative model (i.e., ergodic HMM) at a global level for normal motion patterns. In their work, PCA is employed on the flow vectors to obtain a reduced representation of the flow field during learning stage. Further, only top eigenvectors are selected as representative features and spectral clustering was performed to obtain feature vectors. The HMM model is trained using these features to classify the distinct crowd behaviors in the underlying scene. The results are demonstrated on synthetic simulation on the top view filmed sequences. With a different perspective, Kratz et al. [90] model the statistics of spatio-temporal gradients (i.e., cuboids) with coupled HMM to characterize the behaviors in dense crowds. In their approach, the motion variations in the cuboids are captured and labeled by crowd behaviors. In this manner, the intrinsic structures are captured, and unusual activities are detected as the statistical outliers.

Modeling the dynamics of pedestrian flow, particularly in crowds has been an active research topic in the field of sociology and behavioral sciences. For this purpose, over years a number of models have been proposed by simulating the individual behaviors in crowds. Among many, Social Force Model (SFM) [91] is widely used and is famous to model the pedestrian behavioral dynamics due to its simplicity and intuitive nature. Taking inspirations from pedestrian behavioral

modeling, Mehran et al. [59] suggested a SFM with the optical flow based particle advection technique and simulate the normal social forces of particles implicitly to detect the deviations from pre-trained parameters. In their work, the particles overlaid on the image are advected with the time-averaged optical flow. Further, these particles are treated as individuals and their estimated interaction forces using the SFM are mapped to obtain the force flow of every pixel in the image. Further, these particle forces are modeled for normal behaviors where a bag of words approach is employed to classify a frame as normal and abnormal. In their later work by Wu et al. [87], trajectories of advected particles are modeled to localize the abnormality. However, the resulting tracks on the test dataset are highly inconsistent as it is difficult to determine the objects at the pixel level and their associations in next frames. In addition, discrimination between usual and unusual events is extremely challenging.

Albio et al. [42] maintains the probabilities of optical flow at corner points and constitutes histograms to detect the deviations and abnormalities on PETS2009 [3] dataset. Instead of segmenting and tracking the objects in crowds, the interest points are detected holistically along with their motion. The detected interest points are refined based on the corresponding flows which are analyzed statistically to extract the events in crowds. In similar context, Benabbas et al. [92] build the online probabilistic models for both density and orientation of flow patterns to detect the crowd activities. They constructed an online mixture of Von Mises distributions to model the direction of the optical flow vectors which reveal the major flows of orientation from the mixtures. Moreover, the mean magnitude of the flow vector is modeled probabilistically. Further, spatio-temporal relationship analysis is performed using the direction model, and directional statistics is used to categorize the events in crowd.

Another work is presented by Chan et al. [93] to holistically model the crowd flow in the scene using the dynamic texture model where Nearest Neighbour (NN) and Support Vector Machines (SVM) are used as classifiers to detect the crowd events. Their approach performs crowd counting based on the regression of holistic (i.e., global) features besides the detection of crowd events using the dynamic texture model to represent holistic motion flow in the video. Similarly, Mahadevan et al. [94] proposed a framework to model the normal dynamics of the crowd using mixtures of the dynamic textures hypotheses where the normalcy models indicate joint representations of appearance and dynamics. In their approach, events of low-probability indicate the temporal anomalies while discriminative saliency equates

the spatial anomalies. Moreover, the likelihood map and saliency map are combined together to produce the final abnormality map and anomalies are detected as outliers under this model with high computational cost. The experiments are demonstrated on a new dataset containing the different definitions of anomalies such as walking in wrong direction or vehicles on walking area.

Discussion

In most of the above approaches, anomaly (i.e., abnormal: both terms used interchangeably, unless specified) is formalized as an outlier detection problem. Therefore, in literature, anomaly detection is treated as a context-sensitive term that merely relies on instantaneous motion features. Moreover, capturing the certain motion properties in situations containing any number of concurrent and sparse human activities are extremely difficult. Therefore, the observed flow field tends to result in uncertain information and leads to the plausible outcome. For example, in coherent crowds (e.g., marathon), the object may move with common dynamics which is relatively easy to model. But many scenes (e.g., shopping centers) contain completely random motion of objects resulting in a complicated dynamics, and it is difficult to model the overall dynamics. Besides, it is also found that for an appropriate modeling of scene, the reliable flow patterns play an important basis for supporting effective detection of anomalies and crowd behaviors.

In the literature, generative modeling approaches [89] [59] (i.e., HMM and LDA) require stringent conditional independence among the observed flow fields for more tractable joint distributions. On the contrary, Mehran et al. [59] assume the particles as individuals but they are not able to track anomaly due to the multiple interacting crowd because it requires the knowledge about physical quantities which make this approach impractical. Further, particles are advected based on flow, so the availability of accurate data (i.e., flow field) is an essential requirement. Unfortunately, none of these approaches [90] [59] [87] consider the uncertainty in the observed optical flow and overlooked the limitations of optical flow techniques as described by Bruhn et al. [52].

2.6 Related Issues

We have adapted a top to down mechanism in this research and categorized the related issues in similar hierarchy. This section underlined the related issues which

are taken into account during the development of proposed framework for non-crowded and crowded scenes.

2.6.1 Object Segmentation

We have investigated the research done for the segmentation on the basis of developed methodologies, and the specific task that each work is trying to solve. In the following, we have outlined relevant issues:

- The non-recursive approaches work optimally on simple scenes with the small number of objects but these approaches are not scalable to complex situations where the background proximity is not stable, and the objects are moving with the different pace. In addition, the scene could change from time to time (e.g., sudden or slow illumination changes) therefore, the model should be updated constantly to reflect the most current situations. Moreover, the scheme used to update the background model is not feasible (e.g., in [21]) and the selection of a priori thresholds become a difficult option. The resulting detection contains incomplete information which requires an additional post processing operations.

- The analysis of recursive methods reveals that the inherent idea is to update the intensity values of pixels belonging to the estimated background model which is selected according to the difference between the intensity value of the current image and the corresponding value of the background model. These methods work well when objects move continuously, and the background is not cluttered but are not robust in scenes where the objects are either moving slowly or stop during the motion. Another drawback of these algorithms is that they are not general enough and being heavily conditioned on several heuristic choices. To overcome this shortcomings, we have proposed a segmentation approach in Section 3.1, to combine two approaches for solving these limitations.

2.6.2 Feature Selection

The expedient representation of objects is an essential and challenging task during the development of robust tracking and behavior understanding framework for non-crowded scenes. Therefore, we have aimed in earlier sections of this chapter

to investigate the approaches proposed in the direction of visual features. In the following, we have brought some relevant issues and debate the need and use of fusing multiple features to maintain objects identities when performing the object tracking and behavior understanding in non-crowded scenes.

- Color histogram is one of the preferred choices for describing the objects but it does not hold the spatial information. Various researchers have modeled both the spatial position and color using nonparametric models for instance [95], but this modeling results in an additional computational cost. To overcome this shortcoming, we contend that the region specific information using CSC approach can be incorporated and thus fusing both features at the decision level are more practical. However, in CSC approach, it is essential to assign proper thresholding criteria which is learned by hit and try scheme. The non-feasible parameters results in the failure of region growing mechanism in CSC approach.

- Object appearance can be modeled by exploiting primitive shape models. But, its scope is limited to only rigid objects whereas these models are not persistent enough to handle a variety of objects. For example, a person's shape varies significantly in different camera views. Contours of an object defines both rigid and non-rigid objects but modeling the contours of objects with small size can cause the ambiguous representation whereas the computation load is increased when object's size is large.

The expedient representation of observed flow field which uniquely signifies the dynamics of the underlying crowded scene is a daunting task. In aforementioned sections, we have described the related literature of optical flow and outlined the famous modeling approaches. In the light of related research and our analysis, we have pointed out the relevant issues. The main objective is to design the proposed approach keeping these issues under consideration and to address these limitations to mark our contribution in this research domain.

- Modeling the flow with discrete simulation approaches is a very useful numerical tool, for practical applications but as a research tool, it suffers from the lack of analytical tractability [59]. For example, simulation community performs manual studies to assign the floor field forces based on the scene. But, if the floor field is directly relied on optical flow, it will lack the actual

theme of CA approach and does not result in trustable analysis. Similarly, the SFM suffers from these limitations.

- There are many assumptions which are taken into account, such as: 1) objects have a common sense of task, and 2) objects try to minimize their estimated travel time and are very constraint to the context of application. Due to these reasons, if the model is designed for single objects, it will not be able to handle situations where multiple objects are interacting. Another point, inherent limitation is the requirement of accurate flow field of the crowd. Unfortunately, there are no reliable means to measure such physical quantities using the video data, which makes these approaches impractical for the general scenes. Moreover, the parametric modeling approaches suffered with the computational cost of dense flow field. Therefore, to address the issues of computation, we have applied the modeling at the local level instead of global or holistic level [59].

2.6.3 Tracking and Behavior Understanding in Non-Crowded Scenes

In Section 2.4, we have discussed the related literature and based on our analysis, we have pointed out some key issues which are crucial to address in the proposed approach.

- In data association, uncertainty and ambiguity are two key issues which are usually observed under conflicted situations for instance, data cluttering, and occlusion. In both situations, the flow of information (i.e., the visual characteristics of an object) is intruded, and the data association approach is unable to measure the correct association about the corresponding objects because the likelihood (i.e., similarity) is measured by taking into account the visual features. So, in the situation when the full and long occlusions are observed, tracking performance suffers considerably, and it can even become totally inefficient when discontinuities are inherent in the video.

- The object identity recognition approach can be incorporated for tracking but it is essential to revise the main objectives (i.e., recognition and categorization of object type). Moreover, high quality images are required for the object identity recognition whereas this condition is often violated in object tracking scenarios.

- Logical reasoning can provide a powerful mechanism to determine consistent behavioral states in tracking mechanism and complement its performance. However, it is essential to understand the conceptual structure of possible situations (i.e., object tracking) prior to formulate and design the axioms to infer and address the problem of uncertain and ambiguous interpretations (i.e., in data association) under occlusion and cluttered situations.

2.6.4 Behavior Understanding in Crowded Scenes

In Section 2.5, we have reviewed the literature suggested in the domain of behavior understanding for crowded scenes. Based on our analysis, the underlined issues that are taken into account are:

- Due to the individuals complex interaction in the crowded scene, the computed features such as interest points, localized heads, and specific human classifiers become unreliable. Researchers have employed optical flow, which provides the key clue about objects motion. However, capturing the motion is a essential task in the crowded scenes that contains any number of concurrent and sparse human activities. Unfortunately, many suggested approaches for crowd behavior understanding merely focused on the modeling of flow field at the higher level thus the uncertainties in optical flow measurements are overlooked.

- Another important issue is the definition of crowd behaviors. In most of the above approaches, the anomaly is treated as an outlier detection problem in one class labeling mechanism. This type of rough labeling helps in pinpointing the abnormal locations. However, this assumption is valid for specific situations and handles only one type of behavior (i.e., abnormal). For this reason, it is essential to define the term behavior and its interpretation. Besides, it is also crucial to model the flow field in a manner that represents the distinct flow pattern to play an important basis for supporting effective detection of anomalies and crowd behaviors.

- The classification approaches used in the literature, such as generative modeling approaches [89] and linear modeling [59] (i.e., HMM and LDA) require stringent conditional independence among the observed flow fields for more tractable joint distributions. In crowd behavior analysis where optical flow

is used as the key feature, it is difficult to maintain the conditional independence because, the flow field is significantly correlated during self-evolving dynamic of crowds.

2.7 Discussion and Conclusion

In this chapter, we have presented an extensive survey of tracking and behavior understanding methods and also give a detail insight of related issues. The survey is divided in top to down hierarchy which we have used in this research. First, recognizing the importance of low level processing of the scene, such as object segmentation, and visual features for tracking and behavior understanding systems, we have included a survey of popular object segmentation and feature selection methods. Second, we have provided the detailed review of literature, including the discussion on the data association, object identity management and cognitive modeling, employed by the tracking and behavior understanding algorithms for non-crowded scenes. Third, we have reviewed the approaches devised for crowd behavior understanding along with their categorical discussion and analysis. In the last part of this chapter, we have underlined the related issues which are taken into account in the proposed framework. We believe that, this survey on behavior understanding and tracking for non-crowded and crowded scenes with a rich bibliography content, gives an adequate insight to the readers in this important research topic and encourages new research.

CHAPTER 3

Segmentation

In computer vision, the direct processing of video sequences is computationally expensive, so the first issue is the compact representation of the interesting objects by segmentation. In this chapter, based on the underlined related issues in Section 2.6.1, we aspire to describe the suggested approach for the segmentation in Section 3.1. The results are demonstrated on PETS2006 [1] dataset and comparative studies are conducted to prove the effectiveness of proposed approach for segmentation. Section 3.2 concludes this chapter with a summary and its applicability in the later chapters.

3.1 Weighted Integrated Segmentation Approach

In this section, we have described the algorithm for segmentation by employing the idea of weighted integration of the binarized segmentation responses acquired from different approaches to extract the final foreground under complex situations. Figure 3.1 shows the integration of binary segmentation outcomes of Approximated Median Filter (AMF) [22] approach and Adaptive Background Mixtures Model (ABMM) [26] approach at the detection level, making our approach more robust against scene disturbances and is able to avoid the false detections due to small movements of the surroundings. The motivations are based on two reasons:

- instantaneous variation in the background over time in the scene;
- unexpected behaviors of objects, such as walk, run, and stop in the scene.

The main idea of this work is the evaluation of the intensity variations obtained by both ABMM [26] and AMF [22] approaches through weighted logical constraint, and the final segmentation is achieved by integrating these approaches. In this way, the algorithm is unaffected by the problems of erroneous detection of static pixels and ghost regions in the image when foreground object moves after a long period of time as shown in the results (i.e., ABMM [26] and AMF [22]) in Figure 3.3(f). The entire framework is presented in Figure 3.1 and consists of following steps:

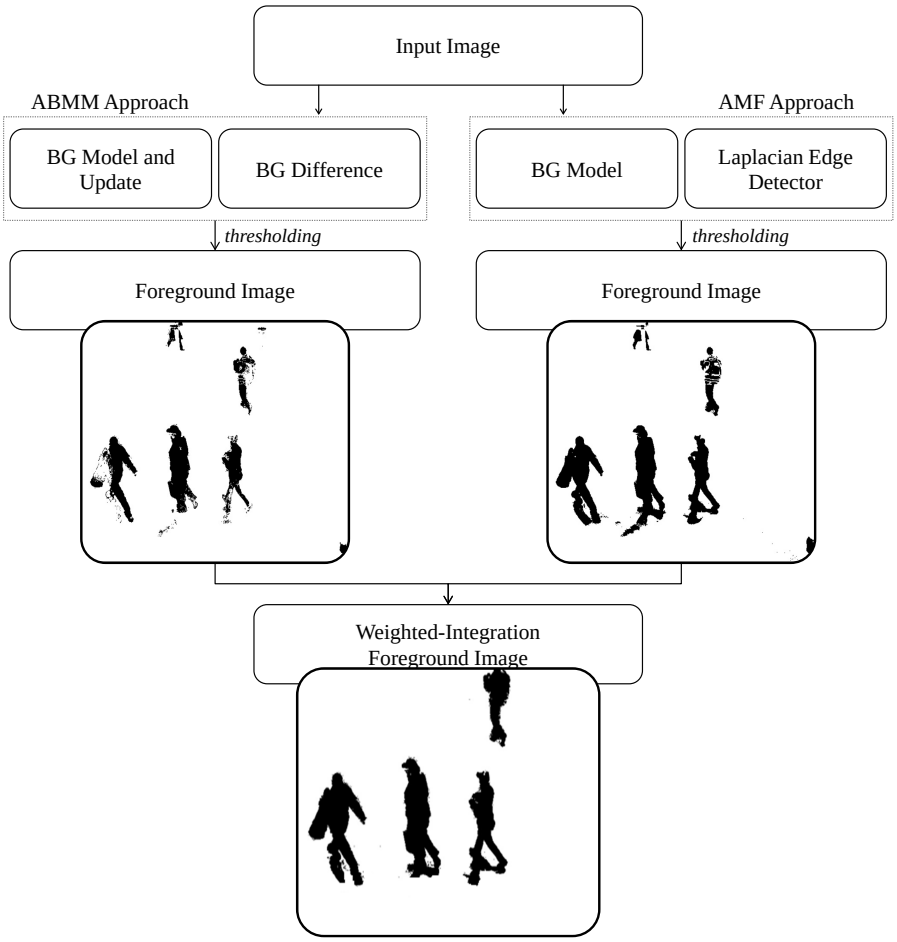


Figure 3.1: shows the process flow diagram of proposed approach for segmentation. The foreground segmentation is performed with ABMM [26] and AMF [22] approach. The segmentation outcome of both approaches are integrated to obtain the final segmentation. Note. for visual display, we have switched the foreground as 0 and background as 1.

3.1.1 Sub-Segmentation by AMF

We have implemented AMF [22] approach in which each successive image is subtracted from a time averaged background model, and the difference image is thresholded. The pixels are labeled as foreground (FG) when they are above the defined threshold. The background (BG) model is estimated by taking a running median of the image sequence, where each pixel in the background model is incremented by one if the corresponding pixel in the current image is greater in value, or decreased by one if the current image pixel is less in value. So, each pixel in the background model is then converged to a median value for which half of the updating values

are greater than, and half are less than the median value. During the update, the background model is assisted by a mask and the corresponding locations around the mask are not updated until the background model is adjusted according to surrounding intensity levels. In this way, this approach handles the situations when the object tends to slow down its motion or stop during the normal walk.

3.1.2 Sub-Segmentation by ABMM

We have implemented ABMM [26] approach for background modeling and foreground detection. The inherent idea is to model the non-stationary temporal distributions of pixels in the video sequence through adaptive mixture of Gaussians. Many variants of this approach have been proposed as explained in Section 2.1. Each pixel in the background model is represented as mixture of Gaussians and an adaptive evaluation[1] is performed at every time step to update the parameters of Gaussian. Two tasks are performed in this approach:

- **Learning the Background Model:** Each pixel in the scene is characterized by the mixture of Gaussians [38]. At each time step, a new pixel value is represented by one of the major components of the mixture models. Given the image sequence I with pixel history $(\{x_1,\ldots,x_t\})$ is modeled by M Gaussian distributions. So, the probability of pixel value is computed by ABMM [26] approach as follows:

$$P(\vec{x}_t, BG+FG) = \sum_{m=1}^{M} \hat{w}_{(m,t)} \mathcal{N}(\vec{x}_t, \hat{\vec{\mu}}_{(m,t)}, \hat{\sigma}^2_{(m,t)} I); \tag{3.1}$$

  where $\hat{\vec{\mu}}_{(m,t)}$ are the estimate of means, $\hat{\sigma}^2_{(m,t)}$ are the estimate of variance, $\hat{w}_{(m,t)}$ are the non-negative weights which are summed up to one, and $\mathcal{N}$ is a Gaussian probability density function. However for computation reasons, the covariance matrix is assumed in ABMM [26] approach as:

$$\Sigma_{(m,t)} = \hat{\sigma}^2_{(m,t)} I; \tag{3.2}$$

  In Equation 3.1, the value of $\vec{x}_t$ for every new pixel is checked against the existing M Gaussian distributions until a match is found[2]. However, we have

[1] A simple heuristic is used to hypothesize the pixels which are most likely to be a part of background learning process.

[2] In the original implementation [26], the value of standard deviation is set to 2.5.

selected the threshold ranging from 3 to 4 standard deviation, empirically. If none of the M distributions are matched with current pixel value, the least probable distribution is replaced with a distribution of the current value as its mean value, an initial high variance, and low prior weight. These weights $\hat{w}_{(m,t)}$ are adjusted in ABMM [26] approach as follows:

$$\hat{w}_{(m,t)} = (1-\alpha)\hat{w}_{(m,t-1)} + \alpha(R_{(m,t)}) \tag{3.3}$$

where α is the learning rate, and $R_{(m,t)}$ is 1 for matched model and 0 for the remaining unmatched models. After this approximation, the weights (i.e., sum of weights is 1) are renormalized.

The μ and σ parameters of unmatched distributions remain the same whereas the parameters of matched distribution are updated with the new observations in ABMM [26] as:

$$\hat{\vec{\mu}}_{(m,t)} = (1-\rho)\hat{\vec{\mu}}_{(m,t-1)} + \rho\vec{x}_t, \tag{3.4}$$

$$\hat{\sigma}^2_{(m,t)} = (1-\rho)\hat{\sigma}^2_{(m,t-1)} + \rho(\vec{x}_t - \hat{\vec{\mu}}_{(m,t)})^T(\vec{x}_t - \hat{\vec{\mu}}_{(m,t)}), \tag{3.5}$$

where $\rho = \alpha\eta(\vec{x}_t|\hat{\vec{\mu}}_{(m,t)}, \hat{\sigma}_{(m,t)})$ is the learning factor which enables the adaptive characteristics, α is used to limit the influence factor of previous data.

- **Classifying Pixels:** After learning the parameters, mixture of Gaussians representing each pixel are sorted down according to their weights in descending topology and the minor weights are discarded. This process scrutinizes the mixture of Gaussians for each pixel according to most likely background distributions. As a result, lower transient background distributions are replaced by new distributions. The resulting distributions are chosen as the background model B_{ABMM} in ABMM [26] as follows:

$$B_{ABMM} = argmin_b(\Sigma^b_{m=1}\hat{w}_{(m,t)} > V_{th}) \tag{3.6}$$

where V_{th} defines the criteria for background subtraction.

Indeed, the ABMM [26] has been employed in real-time surveillance systems for background subtraction and object tracking. This approach is flexible enough to handle variations in lighting, moving scene, multiple moving objects, and other arbitrary changes in the observed scene, but the performance is degraded when the

object stops gradually.

3.1.3 Foreground Detection by Weighted Integration

The segmentation outcomes from both approaches are combined to obtain the final segmentation result by applying the weighted conditional similarity operator which analyzes true and false detections. Let f_{AMF} is the segmented binary outcome of AMF [22] approach and f_{ABMM} is the binary outcome of ABMM [26] approach as shown in Figure 3.2. Mathematically, we can express the combined segmentation f_{intg} as follows:

$$f_{intg}(x,y) = (w_1 \wedge f_{AMF}(x,y)) \vee (w_2 \wedge f_{ABMM}(x,y)) \tag{3.7}$$

Both the weights w_1 and w_2 are measured by applying median filter (i.e., non-linear filter) on the segmented binary pixels of $f_{AMF}(x,y)$ and $f_{ABMM}(x,y)$, respectively. In median filter, a kernel of fixed size (i.e., 3×3, 5×5, or 7×7) is applied over the binary outcome (i.e., $f_{AMF}(x,y)$ and $f_{ABMM}(x,y)$) in a moving window mechanism. The median value in the window is computed for each of center pixel (i.e., f_{AMF} or f_{ABMM}) and results in a binary outcome[3] ($w_1 = 0|1$ and $w_2 = 0|1$). The size of median filter is selected after conducting empirical studies where filter of size (5×5) gives optimal performance. These weights set the criteria which defines level of confidence[4] for the foreground pixel. In this way, the foreground pixels (i.e., labelled as 1 in actual segmentation outcome) are obtained by these two approaches whereas the median-filter based weights enables us to measure the confidence level of classified foreground pixel by considering the characteristics of neighboring pixels.

In summary, both approaches require a single background model (i.e., non-recursive background model) and therefore, the computational performance is quite optimal. The AMF [22] approach has shown better segmentation than ABMM [26] in situations where the object motion is not continuous. However, the precision outcome in AMF [22] approach is effected due to over segmentation whereas in ABMM [26], the segmentation outcome is suffered due to the generated holes and fragmentation. In contrast, ABMM [26] outperforms when the objects are in conti-

[3] As the logical terms are used to obtain final foreground pixel, therefore binary terms are interpreted here as ($0 \longrightarrow false$) and ($1 \longrightarrow true$).

[4] Whether the selected pixel is in actual a foreground or not. The level of confidence is also a binary value which is either 0 or 1.

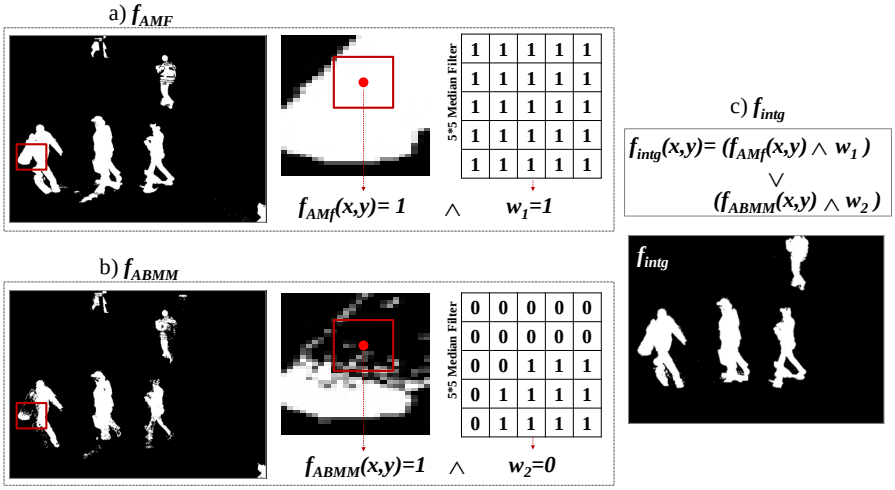


Figure 3.2: shows the process flow diagram of weighted integrated segmentation on a sample frame of PETS2006 [1] dataset. a) shows the binary segmented outcome (f_{AMF}) of AMF [22] approach and the computed binary weight with median filter. b) shows the binary segmentation outcome of (f_{ABMM}) with ABMM [26] and the computed binary weight with median filter. c) shows the integrated segmentation outcome (f_{intg}) of both approaches and the final segmentation.

nuous motion but the performance is degraded when objects gradually or suddenly stops. Therefore, the idea of combining both approaches by weighted integration complements the final segmentation process, and detects foreground under diverse situations. In the following section, results are presented on challenging datasets with the discussion.

3.1.4 Experimental Results and Analysis

The proposed approach is tested on video sequences taken from PETS2006 [1] dataset which represents unique challenges, for example, strong reflections, light variations, and shadows. For evaluation, we have demonstrated a comparative analysis on classical approaches, such as Moving Difference Image (MDI) approach, Running Gaussian Average (RGA) background modeling approach [27], AMF [22] and ABMM [26]. Figure 3.3 demonstrates qualitatively the segmentation on the test PETS2006 [1] sequence where the objects are moving with varying directions and pace. However, for quantitative analysis, ground truth of segmentation is a necessary requirement which is computed manually on selected frames of the sequence as shown in Figure 3.3(a).

In Figure 3.3(b), it is observed that the MDI approach is not appropriate be-

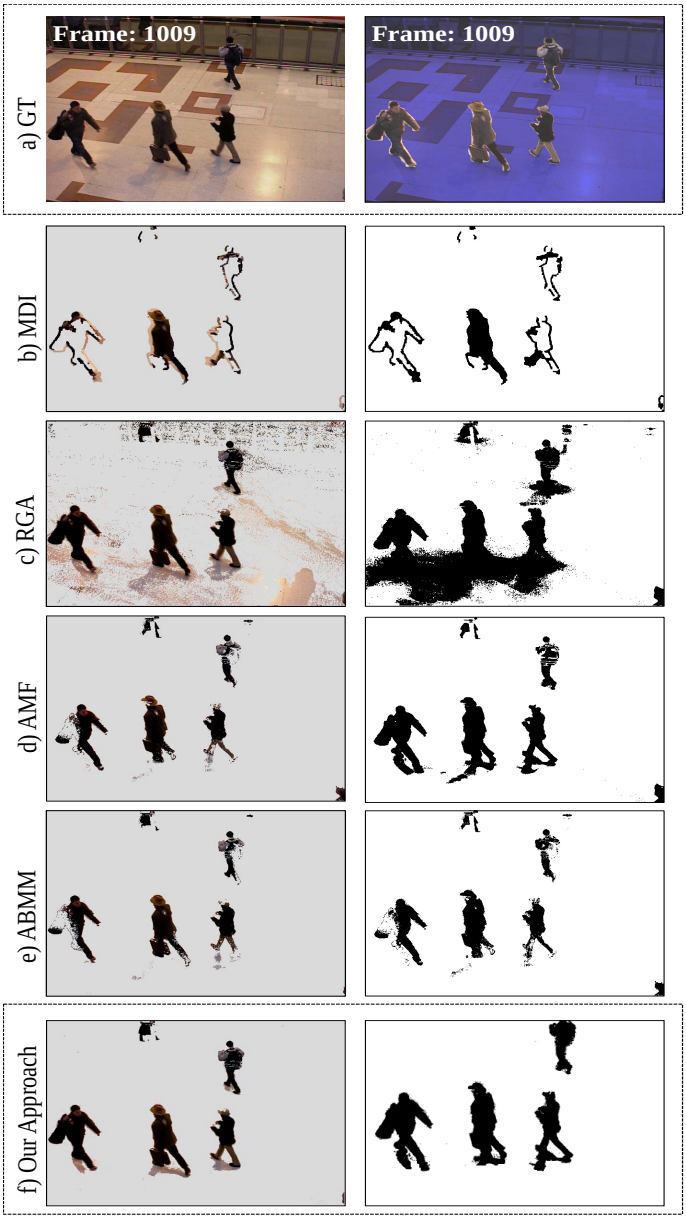


Figure 3.3: shows the foreground segmentation on sample frames from PETS2006 [1] dataset. a) presents the manually created ground truth on test frames. (b-e) present the colored and binary results of state of the art approaches, such as Moving Difference Image (MDI) approach, Running Gaussian Average (RGA) background modeling approach [27], Approximated Median Filter [22] background modeling approach, and Adaptive Background Mixtures Model [26] approach, respectively. f) demonstrates the colored and binary results of Weighted integrated segmentation approach. Note. for visual display, we have switched the foreground as 0 and background as 1.

cause it relies on two factors: 1) prominent distance among frames, and 2) post-processing operations on detected foreground. In the test sequence, objects are moving with varying dynamics and the inter-frame difference is very small. Due to these reasons, MDI approach results in poor segmentation. In Figure 3.3(c), the RGA [27] approach is better than MDI, but the background modeling through averaging is unable to handle the background which contains strong light reflections. Figure 3.3(d) shows the results of AMF [22] approach where background updation is independent of the foreground pixels so it is very robust against moving objects. But, the only drawback is that it adapts slowly towards the large changes in background and needs some additional frames to learn the new background, for instance when the objects in the scene starts moving after being stationary for a long time. The ABMM [26] results are shown in Figure 3.3(e), it can be seen that the objects moving with varying pace are segmented correctly with some under-segmentation issues. However, ABMM [26] approach shows its own drawbacks, such as parameters require careful tuning and it is very sensitive to sudden changes in global illumination. So, if a scene remains stationary for a long period of time, the variances of the background components may become very small. Consequently, a sudden change in global illumination can then turn the entire frame into foreground. Besides, if object suddenly stops while loitering in the scene, it gradually becomes the part of background model.

Figure 3.3(f) shows the result of segmentation obtained by weighted integration approach where the segmentation is preserved regardless of incoherent appearances. Finally, the performance is evaluated by computing precision and recall measures. In the context of binarized segmentation (i.e., 1 refer to background and 0 refer to foreground pixel for visual representation) , precision and recall measures are defined as follows:

$$precision = \frac{Number\ of\ correct\ background\ or\ foreground pixels}{Number\ of\ established\ background\ or\ foreground pixels}, \tag{3.8}$$

$$recall = \frac{Number\ of\ correct\ background\ or\ foreground pixels}{Number\ of\ actual\ background\ or\ foreground pixels}, \tag{3.9}$$

where *actual background or foreground pixel* denotes the background or foreground pixels available in the ground truth.

In Table 3.1, based on the computed ground truth and the segmentation outcome, the precision and recall are computed to demonstrate the quantitative performance. It is notable that both RGA [27] and AMF [22] approaches have high

recall but low precision whereas ABMM [26] approach shows similar values. The performance of MDI is poor among all. The results show efficiency in both the qualitative and quantitative performance of proposed approach over state of the art approaches and this enables us to extract the objects efficiently under various situations. The difference in the performance is more pronounced when shadows are observed or when the objects are moving with different paces over time.

Table 3.1: Comparative analysis of segmentation approaches for PETS2006 [1] dataset in Figure 3.3

Techniques	Precision	Recall
MDI	0.40	0.39
RGA [27]	0.70	0.80
AMF [22]	0.75	0.89
ABMM [26]	0.85	0.85
Our Approach	0.90	0.92

3.2 Discussion and Conclusion

This chapter aims to describe the methodology developed for segmentation within the context of object detection in video sequences. The significance of this chapter is that most of the high level vision tasks, such as tracking, and behaviors understanding, still rely on low level information. Our proposed method is geared towards addressing some of the limitations of existing methods where the comparative analysis provides a clear distinction in performance. Though, segmentation is not a direct objective of this dissertation, but it is a compulsory step to begin any high level vision tasks, particularly related to tracking and behavior understanding. The strength of our proposed approach lies in the fact that it combines the capabilities of two segmentation approaches to ensure good segmentation under diversified situations.

CHAPTER 4

Visual Features

In this chapter, we present visual features that are employed by the proposed behavior understanding approaches for non-crowded and crowded scenes. Section 4.1 describes the idea of ellipse histogram and color-patches of objects. Section 4.2 presents optical flow approach and the fundamental concept of modeling flow field with mixture of Gaussians. Section 4.3 concludes this chapter with discussion.

4.1 Features for Non-Crowded Scenes

The expedient representation of objects is essential to develop robust tracking and behavior understanding algorithm for non-crowded scenes. A number of practical situations are taken into account, and it is found that multiple features can influence substantially in the performance of object matching[1] [96]. In the first color based approach, we have computed an ellipse around the detected object and built an ellipse histogram based on color of the object pixels. In the second approach, we have exploited CSC approach which segments the object into color segments referred as color-patches. Besides, we have also taken into account the object's geometrical features. Our object feature set (f_{feat}) is defined as:

$$f_{feat} = (\varepsilon_h, \zeta_p, area, bb); \tag{4.1}$$

where ε_h is normalized ellipse histogram, ζ_p shows the color-patches of object, *area* defines the area, and *bb* represents the bounding box of object.

4.1.1 Ellipse Histogram

Histogram shows the distribution of color into a calculable form which is used in our Bayesian feature fusion approach in Section 5.4.1. In the computation of ellipse histogram, the main idea of computing histogram remains the same; however, instead of computing the object's histogram directly, we have taken into account the region for computing a histogram bounded by the ellipse as shown in Figure 4.1.

[1] We have exploited two approaches of object's color characteristics to define the feature of the object in our proposed Bayesian Matching Weight (BMW) approach (in Section 5.4.1).

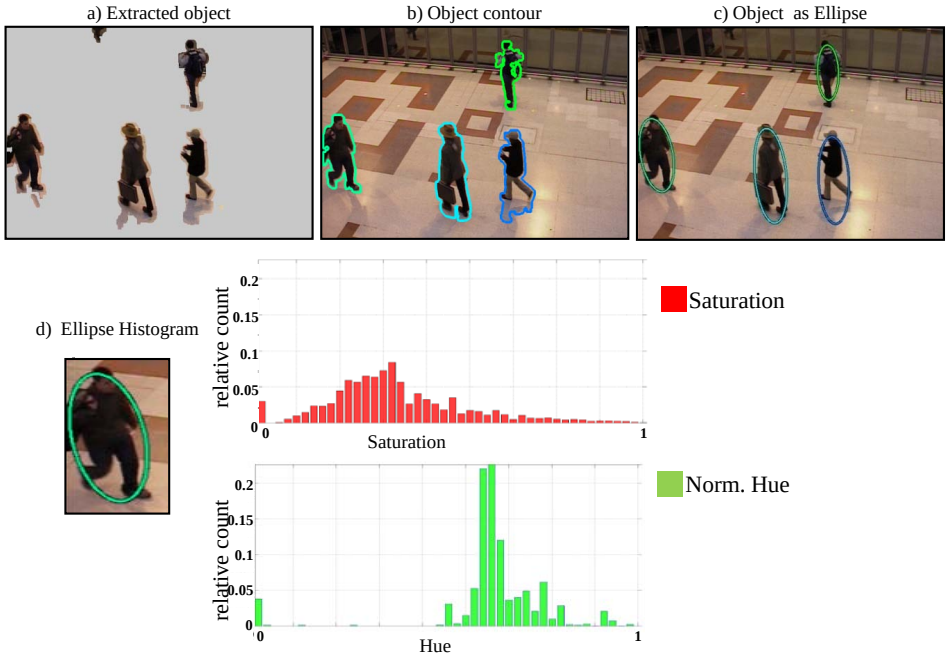


Figure 4.1: shows the process of computing ellipse histogram on a sample frame of PETS2006 [1] dataset. a) shows the extracted objects through segmentation, b) presents the computed contours around the objects, c) represents the computed ellipse over objects along with Hue and Saturation histogram of selected object. The histograms of normalized Hue (it ranges from 0 to 1 when normalized by 360 degrees) and Saturation (it ranges from 0 to 1) channels are formed by dividing these values into 45 discrete intervals (along x-axis) whereas relative count (along y-axis) presents number of pixel counts where the factor of count is obtained by normalizing with number of pixels in ellipse.

In the first, prior to fitting the ellipse around the object, contour of the object is computed as mentioned in Section 2.2.2. After that, we have applied Principal Components Analysis (PCA) [97] algorithm on the contour points of an object. By doing so, a set of Eigen vectors is obtained which is sorted down in ascending manner. The Eigen vector with maximum value is selected which is defined by four parameters, such as height, width, minor angle, and major angle. After obtaining these parameters, the ellipse around the object is computed and its bounded region is extracted. As a result, the detected object is represented by an ellipse whereas the RGB color space is transformed into HSV color space as described in Appendix A.1. The next objective is to compute the histogram for Hue and Saturation channels whereas Value channel is not taken into account. Figure 4.1 shows the histogram of Hue and Saturation channels of objects which is a simple non-

parametric approach where each entry stores the number of pixels of a given color in the data[2]. The x-axis shows histogram intervals with in the range of normalized Hue and Saturation values where the number of intervals is selected empirically[3]. The y-axis shows normalized relative count of pixels in corresponding intervals of Hue and Saturation channels of the object (i.e., bounded by ellipse). In the following, we have conducted experimental analysis on computed ellipse histograms for different color spaces.

Experiments and Analysis

In this section, we have conducted experiments on a sample scene of PETS2006 [1] dataset for determining the effects of color spaces (i.e., gray scale, RGB, and HSV) in the performance of matching processes. Figure 4.2 demonstrates the ellipse histogram computed for each object in the scene whereas Figure 4.3 presents the effects of color space on different histogram similarity[4] approaches as percent matching rate. The values in the graph show the similarity measurement (i.e., in percent matching rate) for each histogram comparison method, such as Histogram Intersection (HI), Euclidean Distance (ED), and Kullback-Leibler (KL) divergence [13] approach. Let the objects o_j are detected at $I(k)$ and o_i are detected at $I(k-1)$, the corresponding normalized ellipse histograms are defined as $o_j(\varepsilon_h)$ and $o_i(\varepsilon_h)$. The formulation of these similarity measurement methods are as follows:

$$HI(o_j(\varepsilon_h), o_i(\varepsilon_h)) = 1 - \sum_{n=1}^{bins} min(o_j(\varepsilon_h(n)), o_i(\varepsilon_h(n))), \tag{4.2}$$

$$ED(o_j(\varepsilon_h), o_i(\varepsilon_h)) = 1 - \sqrt{\sum_{n=1}^{bins} (o_j(\varepsilon_h(n)) - o_i(\varepsilon_h(n)))^2}, \tag{4.3}$$

$$KLD(o_j(\varepsilon_h), o_i(\varepsilon_h)) = 1 - \sum_{n=1}^{bins} o_j(\varepsilon_h(n)) ln \frac{o_j(\varepsilon_h(n))}{o_i(\varepsilon_h(n))}, \tag{4.4}$$

The values of $HI \in [0,1]$, $ED \in [0,1]$ and $KLD \in [0,1]$ are non-negative values where it gives one if the normalized ellipse histograms match exactly and zero if

[2]The image region bounded by an ellipse. However, it can be any region of interest.

[3]Based on the analysis presented in [98], we have selected 45 number of intervals for our histograms.

[4]Histogram similarity is a method of measuring the similarity between the histogram of a reference object at $k-1$ and the histogram of the target object at k.

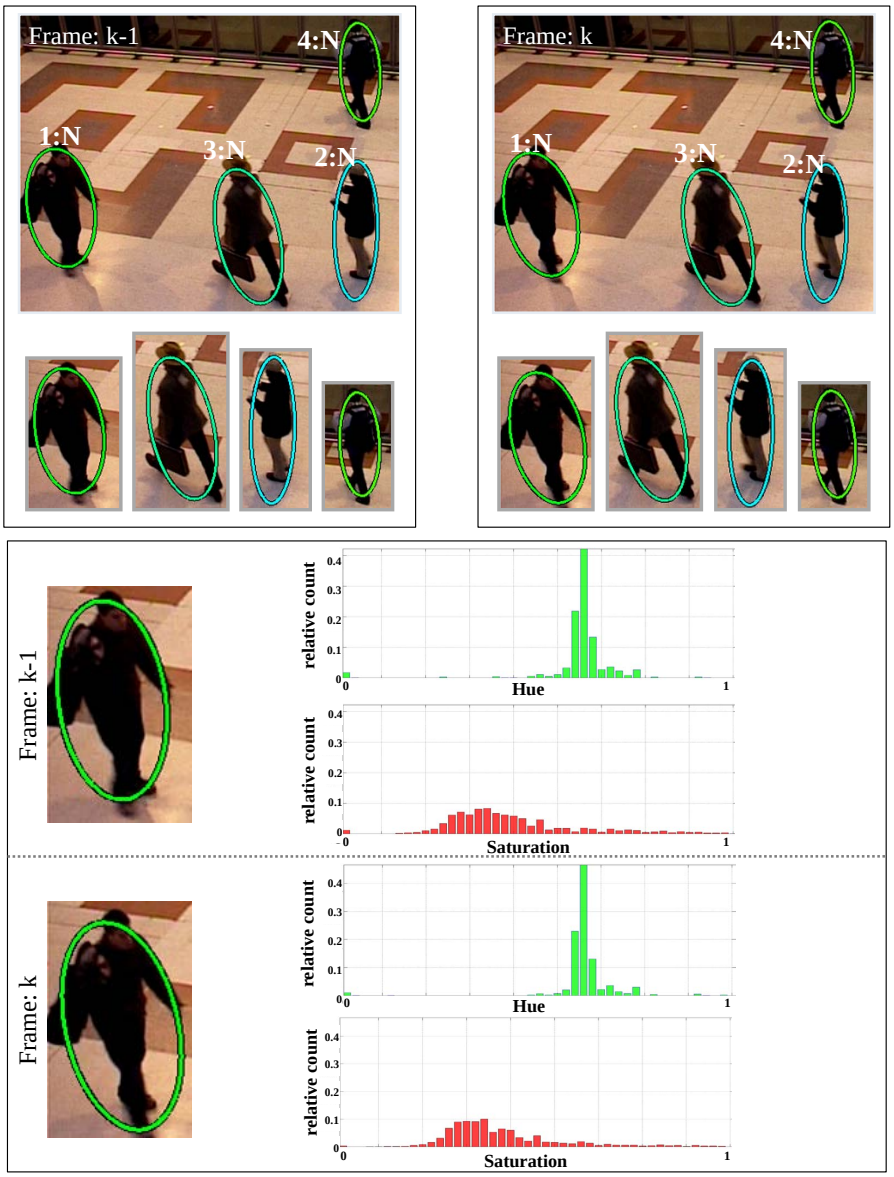


Figure 4.2: shows Hue and Saturation histograms of object on a sample frame of PETS2006 [1] dataset. There are five objects in the scene, and histograms of normalized Hue (ranging from 0 to 1 when normalized by 360 degrees) and Saturation (ranges from 0 to 1) channel are formed by dividing these values into 45 discrete intervals (along x-axis) whereas normalized relative count (along y-axis) represents number of pixel counts.

normalized ellipse histograms do not match.

In Figure 4.3, results indicate that the gray scale color space has lower matching rate and is not able to provide an optimal similarity measure for matching paradigm. In contrast, the similarity results of RGB color space is better when com-

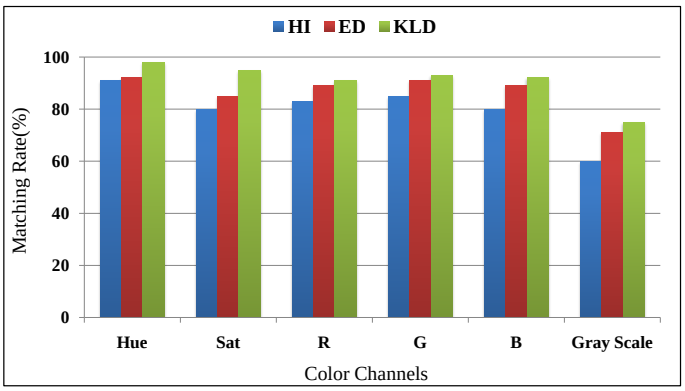


Figure 4.3: presents the results of similarity measures of various color spaces (i.e., gray scale, RGB, and HSV) for each individual color channel with different approaches (i.e., HI in Equation 4.2, ED in Equation 4.3, KL divergence in Equation 4.4) on a sample frame of PETS2006 [1] sequence in Figure 4.2. The graph indicates that the gray scale color space has lower matching rate whereas the similarity results of RGB color space is better when compared to gray scale color space. It is notable that, the similarity measure of Hue (H) and Saturation (Sat) are higher as compared to RGB color space.

pared to gray scale color space. In contast, both Saturation and Hue layers in HSV color space provide the consistent matching values whereas Hue layer indicates better matching results when compared to Saturation layer in HSV color space. Besides, the similarity can be measured by utilizing a single layer which contains color information, for instance, the color information in HSV color space is stored in Hue layer. It can be observed that each color space has different color organization mechanism, for instance, both gray scale and RGB color spaces contain brightness information embedded with color channels, so these color spaces can be easily affected by uneven brightness levels over time. In contrast, HSV contains the brightness information in Value layer and is separated from color information. The results indicated that color spaces, such as gray scale and RGB that utilize brightness information in the color layers has consistently lower matching performance when compared to HSV color spaces. Based on the results shown in Figure 4.3, HSV color space has better similarity measure as compared to the gray scale and RGB color spaces. Therefore, we have used HSV color spaces in our algorithm for consistent performance.

The histograms only record the color information and do not contain the spatial information of pixels. Therefore, they are tolerant to camera viewpoint changes and object movements. An alternative way is to use clustering approach in color

space, but it is often ambiguous whereas the statistical methods used to address this problem are computationally expensive. Therefore, we have utilized the CSC approach along with ellipse histogram in our tracking and behavior understanding algorithm and handle the limitations of the color histogram. The color segments of the object obtained from CSC approach are treated as the features and are used in our matching approach to address the issues of correspondence in our tracking and behavior understanding algorithms as described in Section 5.4.

4.1.2 Color Structure Code Approach

The CSC approach [4] is employed as an object descriptor and is incorporated as second feature in fusion mechanism in Section 5.4. Conceptually, CSC [4] approach is an improved region growing method used to segment the object corresponding to its homogeneous region as described in Appendix A.3. Practically, it follows a parallel hierarchical region growing method on a special hexagonal topology. Therefore, the choice of the starting point and the order of processing are not required. However, the performance of CSC approach relies on the values of its parameters, such as thresholding criteria used in linking and splitting phase which we have learnt by empirical experimentation[5]. The CSC object is treated as a matrix of its CSC color-patches. The CSC [4] approach results in generation of N color-patches for each of the corresponding object and it is described as follows:

$$\zeta_p = \left\{ c_n^{k:id}; n = 1,..,N \right\}, \tag{4.5}$$

$$c_n^{k:id} = (c_{area}, c_{\overline{ncolor_{rgb}}}, c_{bb}), \tag{4.6}$$

where ζ_p is set of N color-patches contained by the object, $c_n^{k:id}$ is the n^{th} color-patch with a set of attributes[6], such as c_{area} is the area, $c_{\overline{ncolor_{rgb}}}$ is the normalized RGB mean color of each color-patch, and c_{bb} is the bounding region of the color-patches. In the following, the CSC [4] approach is employed on our test datasets in Figure A.3.

[5]Small threshold results in the formation of small color regions during splitting phase. In contrast, the large threshold refrain splitting phase to divide the linked segments so, it is essential to select optimal threshold by empirical testing. We have selected the thresholds values 30 for PETS2006 [1] and 45 for PETS2009 [3] datasets through empirical studies.

[6]It is possible to measure other attributes, for example color-patch histogram or Eigen vectors. But currently, we are only taking into account the above attributes.

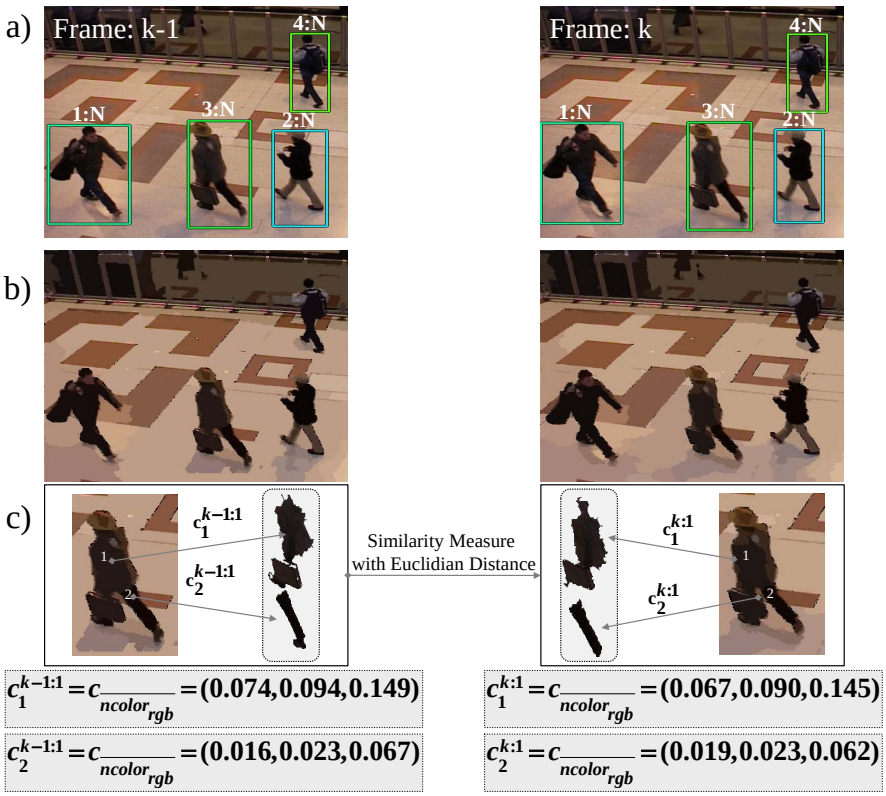


Figure 4.4: a) shows results of CSC [4] approach on a sample frames $k-1$ and k of PETS2006 [1] sequence, b) describes CSC color-patches of object detected at $k-1$ and k and c) shows the normalized RGB mean color $c_{\overline{ncolor_{rgb}}}$ values of color-patches.

Experiments and Analysis

This section presents the experiments and analysis of employing CSC [4] approach to compute CSC color-patches of object as features. The CSC approach results in the formation of color-patches for each object where each color-patch of the object has its attributes, such as area, normalized RGB mean color, and bounding region in Equation 4.6. The similarity among the color-patches of the objects detected at frame $k-1$ and k is computed through Euclidean distance by exploiting the normalized RGB mean color of each color-patch. Moreover, the bounding region c_{bb} is used to define the search space criterion and area c_{area} is used to select the prominent color-patches (i.e., this criterion is empirically selected).

Figure 4.4 demonstrates the results of CSC [4] approach on a test sequence from PETS2006 [1]. It is shown that each object is represented by a set of color-patches. Figure 4.4(c) shows the color-patches of object where the color-patch with

Table 4.1: Similarity measure through Euclidean distance among the object's CSC color-patches at k and $k-1$ on sample frame of PETS2006 [1] sequence

1:N				2:N			3:N			4:N		
$\frac{k}{k-1}$	$c_1^{k:1}$	$c_2^{k:1}$	$c_3^{k:1}$		$c_1^{k:2}$	$c_2^{k:2}$		$c_1^{k:3}$	$c_3^{k:3}$		$c_1^{k:4}$	$c_2^{k:4}$
$c_1^{k-1:1}$	0.99	0.45	0.0	$c_1^{k-1:2}$	0.98	0.41	$c_1^{k-1:3}$	0.99	0.45	$c_1^{k-1:4}$	0.98	0.21
$c_2^{k-1:1}$	0.0	0.98	0.56	$c_2^{k-1:2}$	0.41	0.98	$c_2^{k-1:3}$	0.41	0.99	$c_2^{k-1:4}$	0.21	0.96
$c_3^{k-1:1}$	0.0	0.0	0.95									

prominent area is selected to find the similarity measurement. Moreover, we have also shown the computed normalized RGB mean color of color-patches.

Let the normalized RGB mean color of color-patches $c_n^{k:id}$ of objects detected at k is defined as $c_{\overline{ncolor_{rgb}}} = (r_k, g_k, b_k)$ and normalized RGB mean color of color-patches $c_n^{k-1:id}$ of objects detected at $k-1$ is defined as $c_{\overline{ncolor_{rgb}}} = (r_{k-1}, g_{k-1}, b_{k-1})$. The similarity measure among the color-patches of objects detected at k and $k-1$ is computed as:

$$ED(c_n^{k:id}, c_n^{k-1:id}) = 1 - \frac{1}{\sqrt{3}}\sqrt{(r_k - r_{k-1})^2 + (g_k - g_{k-1})^2 + (b_k - b_{k-1})^2}; \quad (4.7)$$

The outcome of similarity measurement of $ED \in [0,1]$ is a non-negative value where $ED = 1$ represents that color-patches are matched exactly and $ED = 0$ means that color-patches do not match.

Table 4.1 presents similarity measurement results between the normalized RGB mean color of color-patches through Euclidean distance approach. In Table 4.1, it can be observed that the diagonal values are prominently high among the corresponding color patches. For example, object with label $(3:N)$ at frame k containing color-patches $c_1^{k:3}$ and $c_2^{k:3}$ have shown high similarity (i.e., diagonal values) with color-patches $c_1^{k-1:3}$ and $c_2^{k-1:3}$ of object $(3:N)$ at frame $k-1$. Moreover, it is notable that the color-patches acquired from CSC approach provides a consistent representation of object over time as shown in Appendix A.4. Therefore, we have employed the CSC approach as object feature to measure the prior in our BMW algorithm in Section 5.4.1.

4.1.3 Fusing Features in Bayesian Framework

The aim of fusing features (i.e., obtained from different approaches) is to improve the capability of the decision-making process for the tracking and behavior unders-

tanding framework [99]. Typically, there are three possible strategies of feature fusion:

- Early Fusion: Integrating various features extracted from different sources into a single feature vector,
- Cascaded Fusion: Generates the intermediate results considering each feature at a time and makes the final decision based on these intermediate states, and
- Late Fusion: Combine the decisions of different features in a single decision.

A variety of methods are proposed based on fusing features, such as motion and boundary cues [95] [96], integrating Gabor filter along with spatial pixel to support decision [100], combining local and global flow [52], and reasoning about pixels under unusual situations [33]. With similar motivation, we employ the late fusion strategy, and suggest a Bayesian framework to integrate the respective values[7] obtained by ellipse histogram and CSC approach for an object. This fusion is taken place at the decision level, and the objective is to improvise the results and cop the issues discussed in Section 2.6.2. In the following, we have explained the methodology used for object identity assignment problem which we utilize in our tracking and behavior understanding algorithm in Section 5.4.1.

Late Fusion

Late fusion has been successfully used in the domain of biometric system [99] and video segmentation [95]. With similar motivation, we extend this concept to object tracking and behavior understanding problem by utilizing color-based features. Conceptually, the idea behind late fusion is to combine the matching responses acquired from more than one features. For example, if the feature is color histogram, then any suitable histogram matching technique will be used to compute the matching value. In the following, we are intended to describe the fundamental concept of our proposed late fusion approach called Bayesian Matching Weight (BMW) where we have incorporated this idea in our tracking and understanding algorithm as described in Section 5.4.1. Let the detected objects with corresponding features, such as ellipse histogram and color-patches by CSC approach. These detected objects at time k are defined as a set of target $T = \{t_i; i = 1,..,n\}$ and the

[7] The term posterior probabilities, weights or matching response, and other terminologies can be used according to the technique and its implementation.

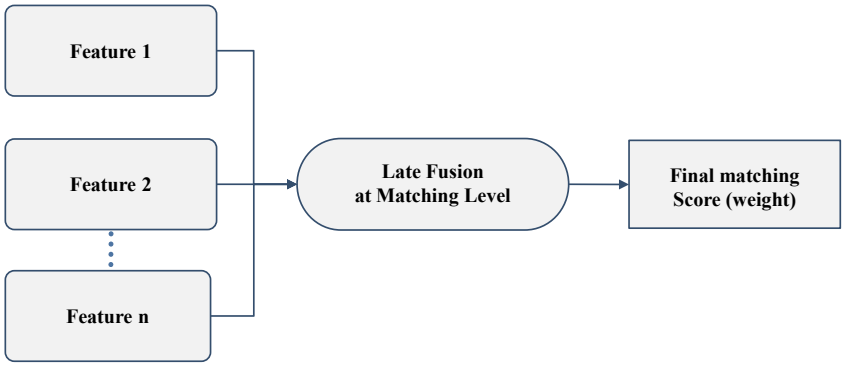


Figure 4.5: The logical diagram of late feature's fusion. In the late fusion the matching responses (i.e., acquired from more than one features) are combined in prior to final matching score.

set of objects detected, identified, and tracked at frame $k-1$ is defined as source $S = \{s_j; j = 1,..,m\}$. Then, the maximum posterior probability [101] of the object s_j corresponding to the target object is denoted as:

$$P_{BMW}(t_i) = \underbrace{argmax}_{t_i} P_{Hist}(t_i|s_j) P_{CSC}(s_j); \tag{4.8}$$

where P_{Hist} and P_{CSC} are the corresponding outcomes from the algorithms employed for ellipse histogram and CSC features. $P_{CSC}(s_j)$ denotes the prior weight assigned to target and $P_{Hist}(t_i|s_j)$ represents the likelihood.

Equation 4.8 describes the fundamental concept of fusion scheme for the decision and Figure 4.5 illustrates the mechanism of fusion. It can be observed that all the features matching outcomes are combined together prior to any decision making and the output of this combination is used for the final decision making process.

4.2 Features for Crowded Scenes

Based on the our analysis and reflections in Section 2.6.2, our interest is to estimate fast and efficient optical flow and used it as a feature to locate the dynamics of distinct crowd block in a scene as explained in Section 6.3. We have employed Anisotropic Huber-L1 [5] method to compute optical flow for video sequences on Graphical Process Unit (GPU) for fast processing. Anisotropic Huber-L1 approach [5] is based on combining data by assuming fundamental constancy of

image property (e.g., brightness) [43] spatially. It is a spatio-temporal regularization approach in which the expected flow across the image is modeled by replacing the isotropic TV regularization with an anisotropic Huber regularization. However, the choice of correct parameters is crucial to obtain optimal optical flow on our test video sequences. In the following, we present the results on test sequences for various optical flow approaches and provide the comparative analysis.

Experiments and Analysis

In this section, we have presented the results of optical flow computed from four state of the approaches. The performance is presented here qualitatively because for quantitative analysis, the ground truth is an essential requirement. However, we provide the computation time of these approaches in frames per second (fps).

Figure 4.6(b) shows the optical flow computed with Horn and Schunck approach whereas Figure 4.6(c) demonstrates the optical flow measured from Lucas and Kanade approach. Moreover, we have also presented in Figure 4.6(d) the computed optical flow with a recently proposed approach[8] by Lie [48]. However, the performance is better but computation time is not feasible. Figure 4.6(e), we have employed Anisotropic Huber-L1 [5] approach on GPU for the optimized processing time. Figure 4.6 shows color-coded optical flows computed from different sequences in our datasets, and it is notable that GPU-based Anisotropic Huber-L1 [5] approach offers good performance and is fairly insensitive to noise.

Mixture of Gaussians for Optical Flow

In this section, we have briefly described the fundamentals of applying parametric modeling approach to obtain a meaningful representation of flow cloud data whereas the detailed explanation is provided in Section 6.5. The optical flow is defined as 2D flow points (i.e., $\vec{f}_p = (v_x, v_y)$) in each selected region of a frame. Since, the observed flow field can be significantly different and correlated; therefore, it is required to glean the information by applying the parametric approximation. The motivation behind using mixture of Gaussians is that it provides theoretically a straightforward way to model our data and forms a comprehensive representation of the flow cloud data inside each selected region.

Our objective is to learn and model the components of mixture of Gaussians over the computed flow cloud data (i.e., $\vec{f}_p$). Given the 2D distribution of flow

[8]This approach is implemented in Matlab so we have developed C++ wrapper for testing.

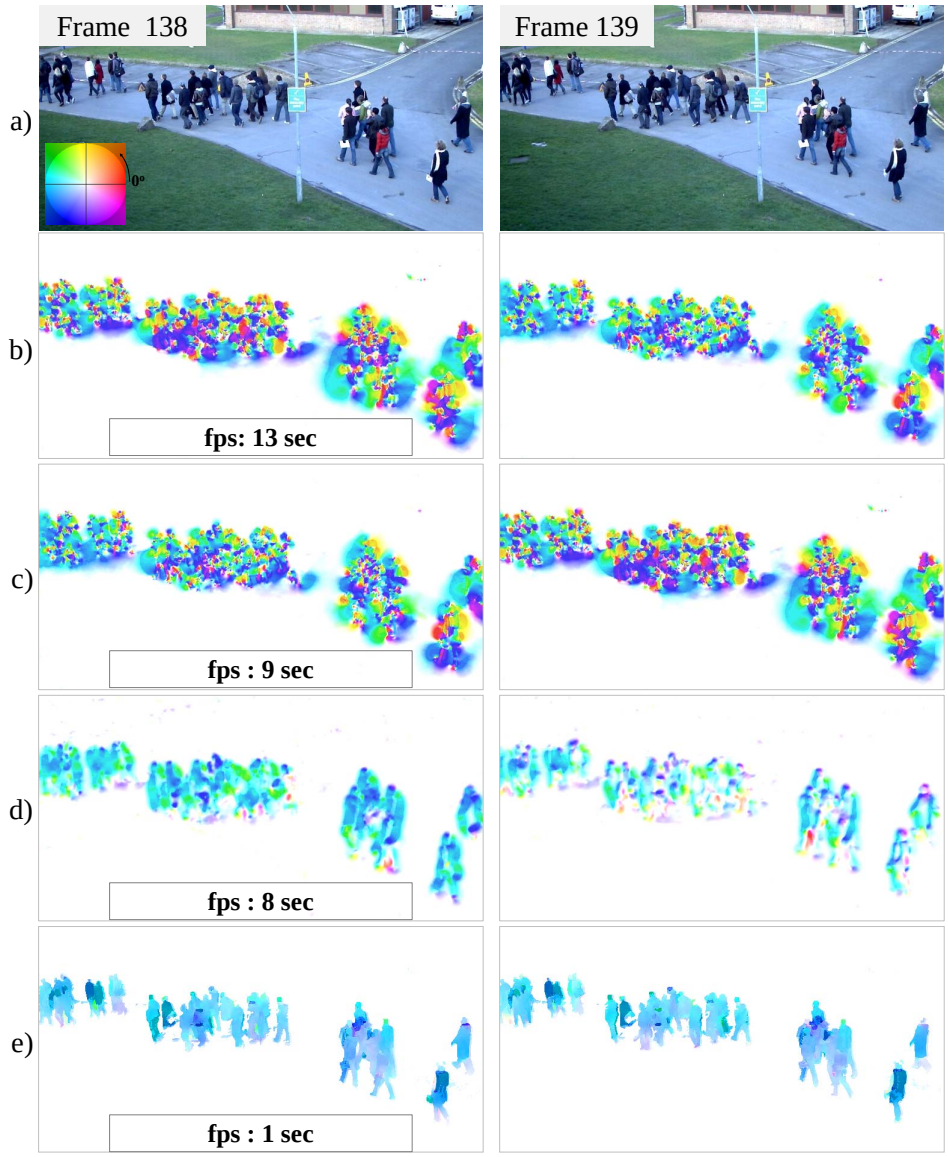


Figure 4.6: shows the computed optical flow with different approaches on the sample frame of PETS2009 [3] along with computation time in frame per second (fps) when executed in Debug mode, b) presents computed optical flow with Horn and Schunck approach [43], c) presents optical flow computed with Lucas and Kanade approach [44], d) shows the result of layered optical flow approach by Lie [48], and e) presents the optical flow measured with Anisotropic Huber-L1 approach [5] on GPU (i.e., nVidia GeForce 9600 GT). The flow field is mapped using the color wheel encoding scheme [102] to indicate its strength.

cloud data in each region as shown in Figure 4.7(a-b), instead to randomly select number of mixtures, we have employed K-means [38] clustering algorithm to

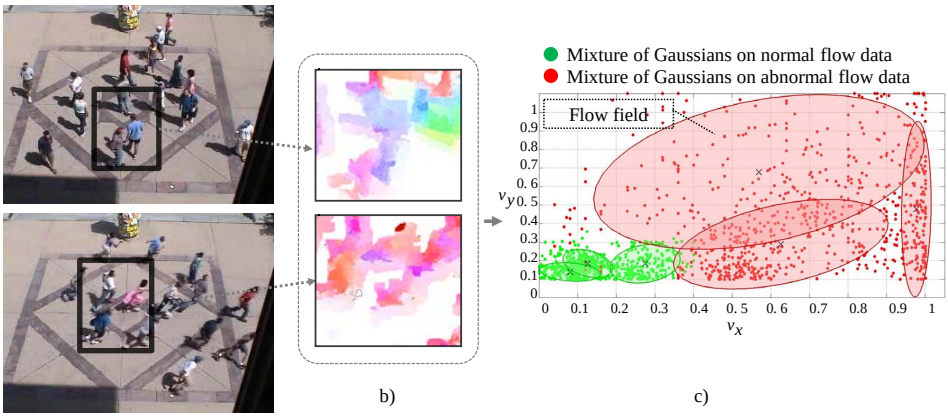


Figure 4.7: shows the process of employing mixture model over observed flow field (i.e., v_x and v_y). a) shows the input image, b) indicates the selected region on which optical flow is computed (i.e., presented in color wheel [102] encoded scheme). c) demonstrates in 2D, the plotted flow field where v_x and v_y are flow velocities in x-direction and y-direction. The mixture of Gaussians is computed from this flow cloud data.

initialize the mixture of Gaussians modeling process. This algorithm returns the number of clusters, and the mixture modeling process is initialized. After that, the Expectation Maximization (EM) [38] is used for finding the maximum likelihood solution for the mixture distributions. Specifically, the parameters of the distributions are estimated to transform the observations into C Gaussian models (i.e., $C = 3$). The Gaussian mixture distribution is written as:

$$p(\vec{f}_p) = \sum_{c=1}^{C} w_c \mathcal{N}(\vec{f}_p | \vec{\mu}_c, \Sigma_c); \tag{4.9}$$

where C represents the Gaussian models, w_c is the weight, $\vec{\mu}_c$ is the mean, Σ_c is covariance matrix which are the components of Gaussian model, and $\mathcal{N}$ is a Gaussian probability density function.

Figure 4.7 shows the flow modeling using mixture of Gaussians on sample frames of UMN [2] dataset. In Figure 4.7(c), the cloud of points shows the flow data where as each ellipse indicates the Gaussian model fitted over the corresponding flow cloud data. There are total six Gaussian distributions over the flow cloud data. In Section 6.5, we have described the modeling flow cloud data in detail for crowd behavior understanding.

4.3 Discussion and Conclusion

In computer vision, most of the problems are ill-posed, therefore, the selection of features and their modeling is essential. In this chapter, we have explained the visual features that are used by the proposed behavior understanding approaches for the non-crowded and crowded scenes. For non-crowded scenes, we have introduced the idea of ellipse histogram and color-patches of the objects for the matching approach. For crowded scenes, we have explained the optical flow approach and the fundamental concept of modeling the flow field with mixture of Gaussians. The aim of this chapter is to explain the relevant features in particular to color, geometrical features, and optical flow in detail due to their relevance to the proposed methods in Chapters 5 and 6, and discussed the possible models used in the vision community.

CHAPTER 5

Tracking and Behavior Understanding in Non-Crowded Scenes

In this chapter, we propose a novel framework for object tracking and behavior understanding by modeling the concepts of human cognitive abilities with statistical approaches for the application of surveillance. Section 5.1 presents the proposed top to down framework for tracking multiple objects and behavior understanding. The framework comprises of four main components presented in Section 5.2, 5.3, 5.4, and 5.5. Section 5.6 demonstrates the experimental results on complex situations and presents that we are not only tracking the objects but also able to successfully infer their behavioral states during motion. The last part in Section 5.7 concludes this chapter.

5.1 The Framework

The proposed framework comprises of four main components: i) object detection through segmentation, and feature extraction, ii) tracking event detection, iii) quantitative and qualitative approach, and iv) tracking system. Figure 5.1 illustrates the relationship between these components.

Segmentation and Feature Extraction: In the first part, segmentation is performed by employing the suggested approach in Section 3.1 and the foreground regions (i.e., objects[1]) are extracted, and the visual features are computed in Section 4.1. The detected objects and corresponding features at each time instance are input to tracking event detection and qualitative and quantitative approach to accomplish the task of high level vision.

[1]The result of segmentation is the foreground region. In the literature [17], the terms blob and objects have been used frequently. Some researchers argument that a blob or region can contain multiple objects in it. But, in this dissertation, we assume that the detected foreground region is object whereas any non-object region is termed as segmentation error.

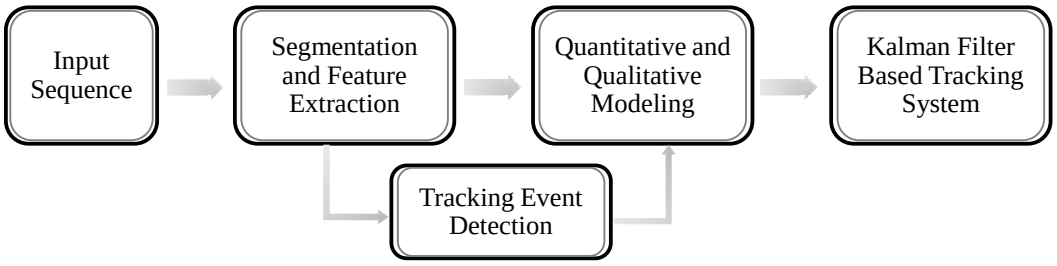


Figure 5.1: The proposed framework: in the first, objects are detected along with their features. In the second approach, tracking events (e.g., occlusion or split) are detected. In the third part, two different information processing approaches (i.e., quantitative and qualitative) are combined and unique identities are assigned to objects. In the last, each object with unique identity is tracked by its respective tracker in Kalman filter-based tracking system.

Tracking Event Detection: There are many events observed during tracking process, such as occlusion, split, new entry, and exit. In this, we have detected these events, and the respective logical functions are triggered.

Quantitative and Qualitative Approach: In this component, we have proposed the two approaches (i.e., quantitative and qualitative) along with their integration. At conceptual level, the overall goal is to track and understand the individual behaviors of objects. In dynamic scenes, identification of objects using typical statistical matching algorithms (i.e., as discussed in Section 2.4.1) normally give very poor results under conflicted situations [70]. This is due to the interactions among moving objects, for instance, occlusions in which the objects overlap or completely hide the other object. It is evident that treating object tracking as the cognitive problem and use it with the typical statistical algorithm can improve the overall tracking performance under non-feasible situations [33].

At practical level, all the detected objects are considered as a node and constitute an undirected graph [103]. Moreover, these objects are described by unique identities (i.e., from Identity Pool) and data structure comprises of quantitative (i.e., visual characteristics) and qualitative (i.e., cognitive characteristics) information at each time instance as illustrated in Figure 5.4. First, the matching weights of the objects are computed. Second, the axioms are developed by employing the principles of human-perception for the tracking process. These axioms assign the behavioral states to detected objects which are associated with the matching weights. Essentially, these two ap-

proaches function together and the behavioral state of an object is inferred by incorporating the reasoning functions while satisfying the fundamental constraints of continuity of objects during tracking.

Tracking System: We have developed a Kalman filter-based tracking system where every tracker is associated with an object and estimates its state over time.

5.2 Segmentation and Feature Extraction

The first task in tracking is to detect objects in a given video sequence which serves as an input for further high level processes. For this purpose, we have suggested a segmentation approach to detect objects in the test sequences as presented in Section 3.1. The next objective is to compute features that depict the unique representation for each object. So, in Section 4.1, we have described our feature set f_{feat} as follows:

$$f_{feat} = (\varepsilon_h, \zeta_p, area, bb); \tag{5.1}$$

where ε_h is the normalized ellipse color histogram, ζ_p is the CSC color-patches, *area* is the objects area, and *bb* defines the object bounding region.

5.3 Tracking Event Detection

Challenges in real-time object tracking are multitudinous whereas highly constrained solutions have been proposed as mentioned in Section 2.4.1. In tracking paradigm, an important issue is to detect efficiently the instances of events[2] associated with tracking. In general, events like occlusion, split, new, exit, etc., are frequently observed in the scenes where the detection of these instances is essential to accomplish the tracking and behavior understanding task. For instance, various approaches [104] [105] [106] directly or indirectly address the detection of these events in tracking paradigm, however, it is very hard to find any concrete solution in the literature.

We have categorized these events into four types of events: 1) new, 2) exit, 3) occlusion, and 4) split. The detection of these events is based on mapping the spatial occupancy of objects at time k with the objects at time $k-1$. So, ideally each object should contain only one object which shares its spatial space in the

[2]The term events refer to occlusion, split, object entry and exit in scene during object motion.

next frame where it is assumed that the objects are moving smoothly. Based on this assumption, a "spatial mapping matrix" of the detected objects at k and $k-1$ is built. Later, the criteria for these events are defined which are deduced based on the spatial mapping matrix[3].

Building Spatial Mapping Matrix

A detected object is a segmented region of input image from a sequence and is defined as a 2D function $I(k) = h(x,y)$ where h indicates the intensity and (x,y) contains the spatial information. We consider the spatial information (x,y) of each detected region at k and $k-1$, for creating the spatial mapping matrix. The number of columns in the matrix is equal to number of the detected objects at k. In contrast, number of rows in the matrix is equal to number of detected objects at $k-1$. Now, the main concept is to map the spatial correspondence of objects whereas the objects that do not share spatial correspondence are excluded. Let us assume that the detected objects at k are $I(k) = \{o_j; j = 1,\ldots,m\}$ and objects at $k-1$ are $I(k-1) = \{o_i; i = 1,\ldots,n\}$. The spatial correspondence is computed by measuring the spatial occupancy which is the ratio of spatial region of an object at k mapped over the spatial region of an object at $k-1$. The area of the spatial region of object at k is represented by $I(k)_{s_a}$ and the area of spatial region of object at $k-1$ is represented by $I(k-1)_{s_a}$. The ratio between the mapped spatial regions is defined as the percentage of spatial mapping S_r which determines the relative spatial occupancy of an object at k and is computed as follows:

$$S_r = \left(\frac{I(k-1)_{s_a}}{I(k)_{s_a}} \right) \times 100; \tag{5.2}$$

where S_r defines the relationship among the objects at k and $k-1$. Based on above spatial relationship quantity, we have developed a set of criterion for each of the tracking event. Moreover, we have demonstrated a test case in Figure 5.2 to provide an insight about each of following criteria to detect the respective events.

- New Event (n_{active}): Each new object entering the scene at k does not share any spatial correspondence with the existing objects at $k-1$. The object does not share any spatial correspondence results in S_r which is either very small or close to zero as shown in Figure 5.2(a). The third column indicates that

[3]This matrix contains fields which are filled based on whether an object shares its spatial space in the next frames or not.

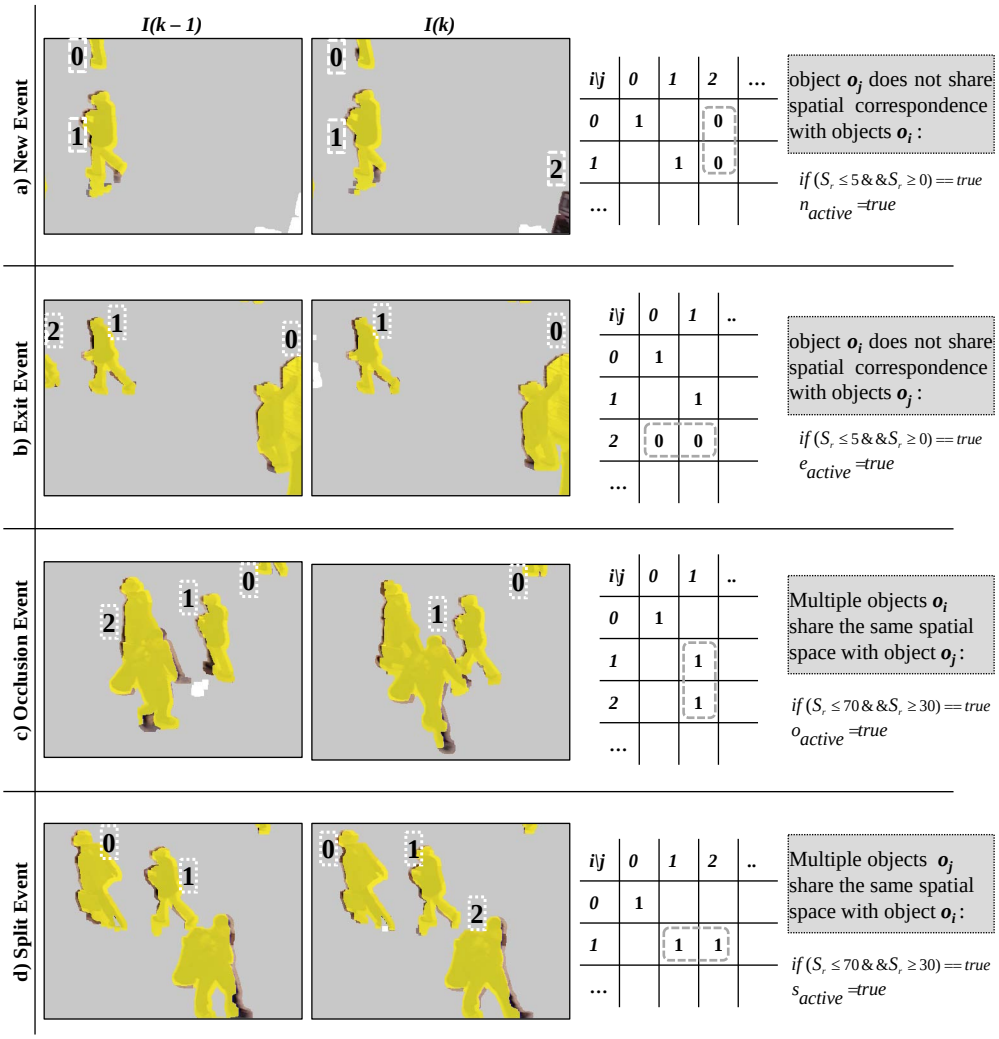


Figure 5.2: presents the concept of tracking event detection where 1 (i.e., true) represents that spatial correspondence of object is found at k and 0 (i.e., false) shows that the object does not find any relationship. The objects are given arbitrary identities to explain the idea. Moreover, objects detected at k frame is represented by its own visual attributes where the yellow transparent information shows the objects at $k-1$. In this figure, a) shows new event, b) represents exit event, c) represents the occlusion event , and d) shows the split event.

no spatial correspondence (i.e., indicated by 0) is found with the object at k and thus n_{active} event becomes true. This condition is formulated as follows:

$$if(S_r \le 5 \;\&\&\; S_r \ge 0) == true$$
$$n_{active} = true$$

- Exit Event (e_{active}): Object which leaves the underlying scene after moving around is said to be disappeared or exiting from the scene. When S_r is computed, it is found that the spatial correspondence of the object with the objects of frame k is missing as indicated by 0 (i.e., in entire third row) in Figure 5.2(b) and e_{active} flag is set. This condition is formulated as follows:

$$if(S_r \leq 5 \;\&\&\; S_r \geq 0) == true$$
$$e_{active} = true$$

- Occlusion Event (o_{active}): The interception of visual attributes results in the phenomenon of occlusion when multiple objects share the same spatial space. The object which shows its spatial occupancy with more than one object at $k-1$ is the merged object as indicated by 1 (i.e., in the entire second row) in Figure 5.2(c). The S_r ranges from 30 to 70%[4] and $o_{active} = true$. This condition is formulated as follows:

$$if(S_r \leq 70 \;\&\&\; S_r \geq 30) == true$$
$$o_{active} = true$$

- Split Event (s_{active}): The end of visual interception among the objects results in split event. This event is observed when more than one object at k shows the spatial relationship with the detected objects at $k-1$ as denoted by 1 (i.e., in the entire second column) in Figure 5.2(d). In this event, S_r ranges from 30 to 70 % and s_{active} is set. This condition is formulated as follows:

$$if(S_r \leq 70 \;\&\&\; S_r \geq 30) == true$$
$$s_{active} = true$$

In this manner, we are able to detect the conflicted events which are linked to quantitative and qualitative approaches.

[4]The percentage is empirically determined by experimental analysis.

5.4 Quantitative and Qualitative Approach

In this section, we have suggested the quantitative and qualitative approaches which are driven by segmentation and feature extraction approaches. The motivation is to tackle the tracking problem by axiomatizing and reasoning the human-tracking abilities as shown in Figure 5.3. At the broader level, each detected object is treated as a node and an undirected network of detected objects is built in spatial space. At the lower level, the object is described with a unique identity, and a data structure comprising of both quantitative (i.e., visual features) and qualitative (i.e., logical features) information. The matching weights are incorporated with the tracking axioms to deduce the appropriate behavioral states of the objects at each time instance. However, during occlusion, the correspondence approaches lead to ambiguities by assigning wrong matching weights due to incomplete visual contents. The qualitative approach handles these situations by employing tracking axioms. For example, if o_{active} is active, then it calls the corresponding developed tracking axioms for occluded and overlaper situations which infer the respective behavioral states (i.e., overlaper or occluded) to the objects, explicitly. Essentially, this mechanism handles the conflicted situations to disambiguate the object's behavioral states and manage identities during tracking. Later, the objects are linked with the tracking system and each object is associated with its respective tracker to estimate the trajectories of these objects.

In quantitative approach, Bayesian inference is employed to measure the posteriori probability of the objects which is referred as matching weights of the objects. The reason of using Bayesian inference is based on its philosophy "...all information about the world is captured by the posterior... [107]". The first justification of this view is that the posterior is actually a combination of prior information about the world and a model of the process by which measurements are generated (i.e., so it covers every aspect and no information is missing from the posterior), and the available information is combined in a proper manner. The second justification of this view can be derived from the first, which is the way the posterior is computed, produces good results. However, it is a great challenge to efficiently compute the posteriors and this defines the first innovation of our work. We have aggregated two different approaches within Bayesian inference. The KL divergence between the objects gives the likelihood of the Bayesian inference whereas the prior information is measured with CSC approach. This methodology works efficiently during conflicted situations which are observed when object's contextual information (i.e.,

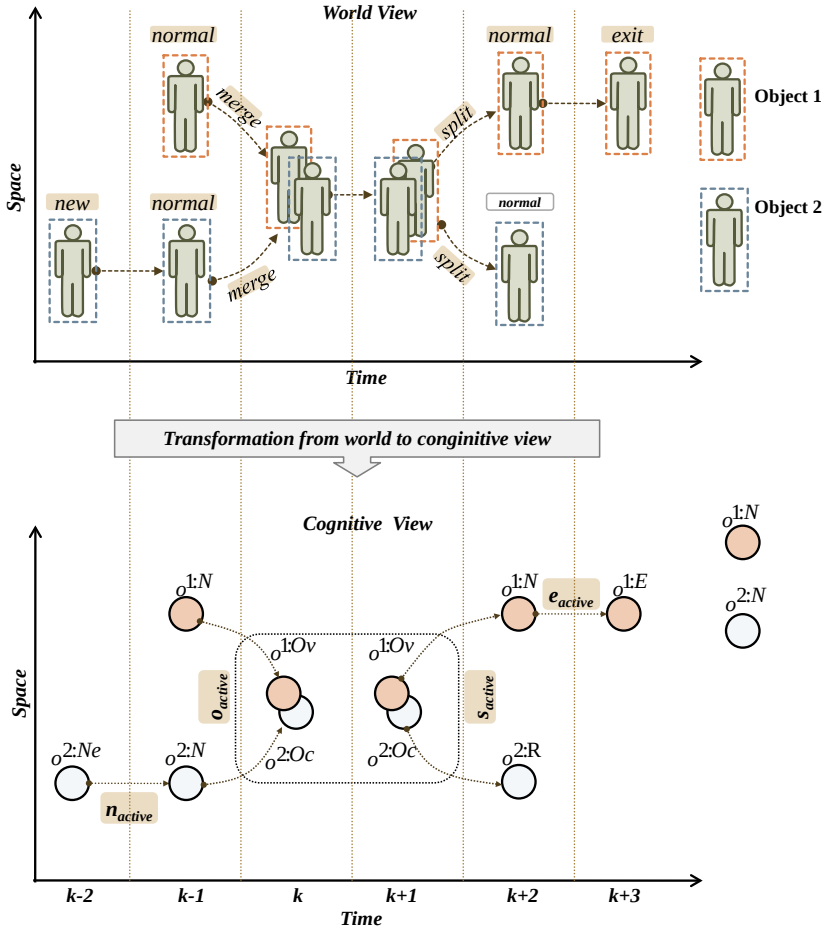


Figure 5.3: From the real scene to the concept, and the representation of an object both in qualitative and quantitative manner are presented. The time instances representing conflicts are highlighted and the transformation of this information into the spatial domain is shown in the logical view. Each detected object is defined by a compound data structure with unique identity, visual features (e.g., normalized ellipse histogram, CSC color-patches, area), and behavioral states (e.g., Normal (*N*), New (*Ne*), Exit (*E*), Overlaper (*Ov*), Occluded (*Oc*), Reappear (*R*)).

visual characteristics) is lost partially.

As the second innovation of this work, an explicit novel qualitative approach is suggested to handle the ambiguities due to occlusions and conflicted situations. The idea is motivated by human's perception, learning, and knowledge acquisition abilities to infer under observation entities and its context. We argument that, the models designed by taking into account these abilities which are usually referred as human cognitive abilities (i.e., these terms will be used interchangeably unless

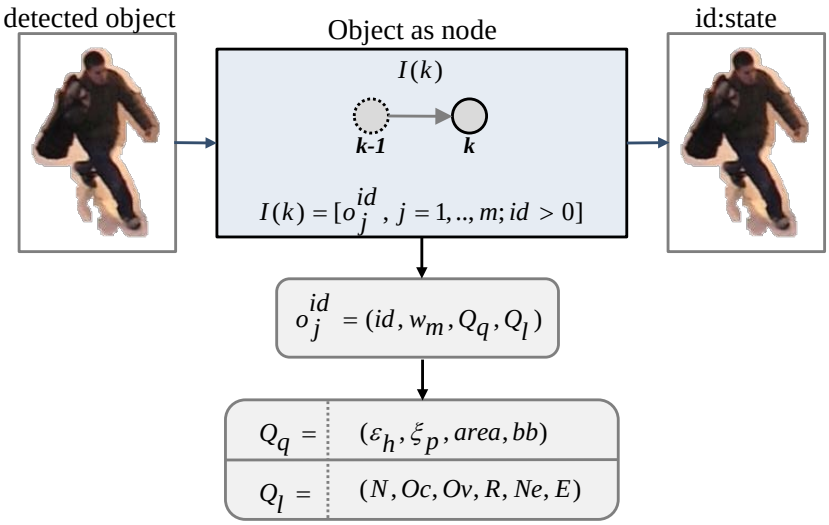


Figure 5.4: shows the illustration of object as node along with its characteristics. $I(k)$ is the image frame at k which contains m detected objects o_j^{id}. The detected objects are represented by their unique identities id, quantitative characteristics Q_q (i.e., visual features) which are processed by quantitative approach to measure the matching weight w_m, and set of behavioral states Q_l which are inferred with qualitative approach.

specified) provide an efficient mechanism to handle the ambiguities in quantitative approaches. In the context of object tracking scenario, first, the problem of tracking and the corresponding behaviors are studied with human perspectives which are then mapped by employing the knowledge of logical modeling with propositional logic [108]. Each behavior is encapsulated with a logical expression named as axioms to make inference about the corresponding behavioral states during motion and to manage the identities. Both quantitative and qualitative algorithms are bi-directionally linked and complement their functionalities by improving the performance dramatically. Moreover, the effects of uncertainty and ambiguity are addressed whereas the object behavioral states are inferred along with object identity management and tracking.

Given a video sequence which is composed of K frames, it is assumed that each detected object is a continuous function of time in the scene until it leaves permanently. The detected object at frame k is:

$$I(k) = \left[o_j^{id}; j = 1, \ldots, m; id > 0\right]; \tag{5.3}$$

where $I(k)$ is the image frame at k time instance, o_j^{id} are the m detected objects

with unique identities *id* as shown in Figure 5.4. Each detected object contains a unique identity *id*, its quantitative characteristics Q_q (i.e., features as mentioned in Section 5.2) which are processed by quantitative approach to measure the matching weight w_m, and set of behavioral states Q_l which are inferred with qualitative approach. So, the object is defined as:

$$o_k^{id} = (id, w_m, Q_q, Q_l); \tag{5.4}$$

Each individual object contains a set of attributes (id, w_m, Q_q, Q_l) where,
id : is the unique identity of object.
k : is the frame number.
Q_q : represents the visual characteristics of the object.

$$Q_q = \left(f_{feat}\right); \tag{5.5}$$

the above expression can be rewritten from Equation 5.1 as:

$$Q_q = (\varepsilon_h, \zeta_p, area, bb); \tag{5.6}$$

Q_l : represents the object behavioral states as described in Table 5.2.

$$Q_l = (N, Oc, Ov, R, Ne, E); \tag{5.7}$$

5.4.1 Quantitative Approach-Bayesian Matching Weight

A multi-layered quantitative[5] algorithm based on Bayesian inference is proposed by formulating the detected object as an undirected graph in Figure 5.3. In this approach, the main objective is to compute the matching weights of object in an efficient manner during object tracking in both ideal and conflicted situations. To achieve this task, we have aggregated two different techniques in Bayesian inference for computing the matching weights (i.e., posterior probability, unless specified) of an object as shown in Figure 5.5. First, the normalized ellipse histogram is approximated around the detected objects at $k-1$ and k. Second, CSC approach is exploited to compute the prior weights of the objects.

Inferencing about an object's possible occurrence at $I(k)$ is made by incorpora-

[5]In Section 4.1.3, we have described the fundamental concept of the proposed matching approach.

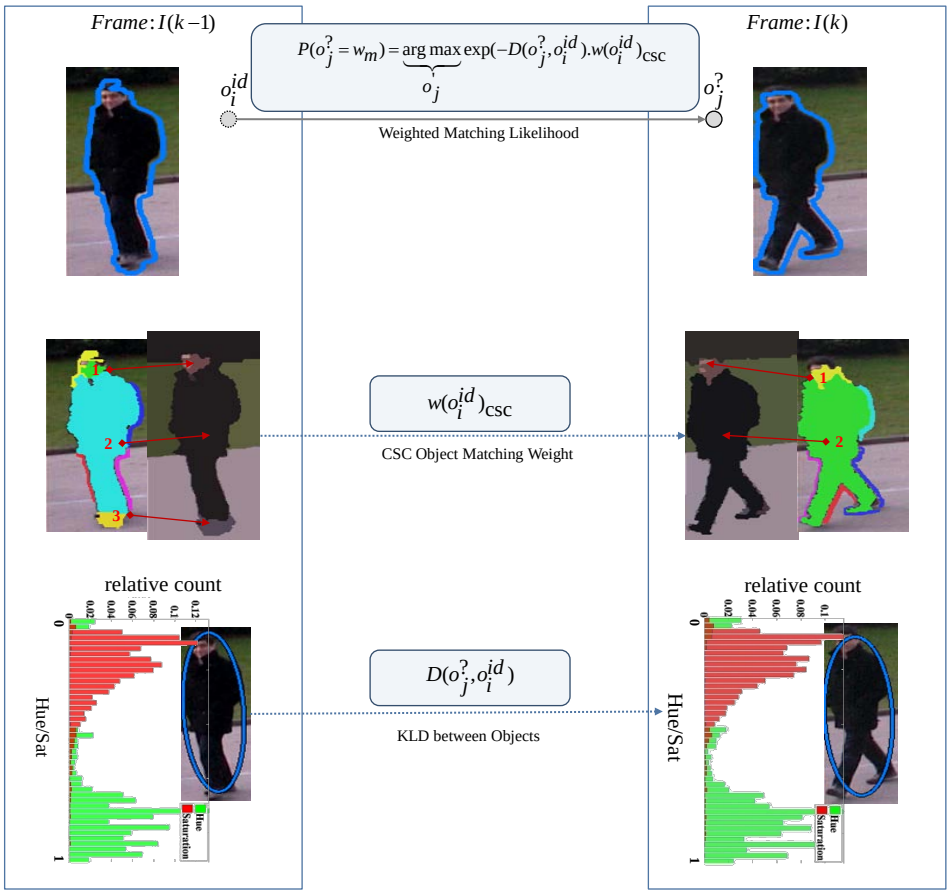


Figure 5.5: shows the computation level of BMW approach on sample frame of PETS2009 [3] dataset. The first level defines the main formulations of computing matching weights w_m among the objects detected at frame k and $k-1$. The second level computes the likelihood through KL divergence $D(o_j^{'}, o_i^{id})$ among the normalized ellipse histogram of detected objects at frame k and $k-1$. Both normalized Hue(i.e., normalization factor of 360 degrees) and Saturation channels values (i.e., ranging from 0 to 1) are divided into 45 intervals. The third level finds the prior weight $w(o_i^{id})_{csc}$ by measuring the Euclidean distance among the color patches of objects using CSC approach.

ting prior probability measured from the possibilities $I(k-1) = \left[o_i^{id}; i = 1,..,n; id > 0\right]$, and the likelihood evidence of the observed data $I(k) = \left[o_j^{'}; j = 1,..,m\right]$. The maximum a posterior probability (MAP) [101] of an object at $I(k)$ corresponding to objects at $I(k-1)$ is computed as:

$$P(o_j^{'} = w_m) = \underbrace{argmax}_{o_j^{'}} P(o_j^{'}|o_i^{id})P(o_i^{id}); \tag{5.8}$$

where $P(o_j^{'}|o_i^{id})$ is the likelihood between objects at k and $k-1$ which is interpreted as KL divergence $D \in [0,1]$, between objects at $I(k-1)$ and $I(k)$. The likelihood computation can be measured as follows:

$$P(o_j^{'}|o_i^{id}) \Rightarrow exp(-D(o_j^{'}, o_i^{id}); \tag{5.9}$$

Similarly, $P(o_i^{id})$ is the objects prior probability at $I(k-1)$ which can be defined as prior weight $w(o_i^{id})_{csc}$ assigned to each object at $I(k-1)$. The objects prior probability is computed as:

$$P(o_i^{id}) \Rightarrow w(o_i^{id})_{csc}; \tag{5.10}$$

We can re-formalized the Equation 5.8 as follows:

$$P(o_j^{'} = w_m) = \underbrace{argmax}_{o_j^{'}}(exp(-D(o_j^{'}, o_i^{id})).w(o_i^{id})_{csc}); \tag{5.11}$$

In the following, we have explained the methodology of measuring the KL divergence D and CSC approach-based prior weight w_{csc}.

Measuring KL Divergence Between Objects

The Kullback-Leibler (KL) divergence measures the similarity matching as minimum cost [13] among the detected object at $I(k)$ and $I(k-1)$. The KL Divergence has the intimate relationship with likelihood theory and it measures the proximity among the object's normalized ellipse histogram. Let the object $o_j^{'}$ is detected at $I(k)$ and object o_i^{id} is detected at $I(k-1)$, the corresponding histograms are defined as $o_j^{'}(\varepsilon_h)$ and $o_i^{id}(\varepsilon_h)$. The KL divergence is defined as:

$$D(o_j^{'}(\varepsilon_h), o_i^{id}(\varepsilon_h)) = \sum_{n=1}^{bins} o_j^{'}(\varepsilon_h(n)) ln \frac{o_j^{'}(\varepsilon_h(n))}{o_i^{id}(\varepsilon_h(n))}; \tag{5.12}$$

where D is non-negative $D \geq 0$, not symmetric in $o_j^{'}(\varepsilon_h)$ and $o_i^{id}(\varepsilon_h(n))$, zero if the histograms match exactly and can potentially equal to infinity but we have treated the infinity as 1 for bounded solution which means that objects have high divergence.

It is worth to mention that several distance measures were considered, and ex-

periments show the effectiveness of the proposed combination of distances as discussed in Section 4.1. Figure 5.6(a-b) illustrates the matching of two objects at k and $k-1$ using KL divergence whereas Table 5.1 indicates matching scores on test sequence.

CSC-Based Object Prior Weight

The prior weight is computed based on how much information an object carries at $I(k-1)$ about the newly observed object at $I(k)$ as shown in Figure 5.6. In other words, given the set of color-patches representing an object as discussed in Section 4.1.2, we have measured the similarity as prior weight[6] and find the most similar color-patches of the objects satisfying the search space criterion. The computed prior weights of color-patches of objects are averaged to measure the final prior weight. We have employed Euclidean distance to measure the similarity among the color-patches of the objects. Let the object $o_j^{'}$ is detected at $I(k)$ contains color-patches $\zeta_p = c_s^{k:'}$, and the object o_i^{id} is detected at $I(k-1)$ contains color-patches $\zeta_p = c_r^{k-1:id}$. So, the prior weight indicates that how much an object detected at $I(k-1)$ contains the contents of objects detected at $I(k)$ and is computed as follows:

$$w(o_i^{id})_{csc} = 1 - \sqrt{\frac{(o_j^{'}(c_1^{k:'}) - o_i^{id}(c_1^{k-1:id}))^2 + \ \ldots \ + (o_j^{'}(c_s^{k:'}) - o_i^{id}(c_r^{k-1:id}))^2}{R}}, \tag{5.13}$$

$$r = \{1,\ldots,R\}, \ \ s = \{1,\ldots,S\};$$

where $c_r^{k-1:id} = c_{\overline{ncolor_{rgb}}}$ contains normalized RGB mean color of R color-patches, $c_s^{k:'} = c_{\overline{ncolor_{rgb}}}$ contains normalized RGB mean color of S color-patches, and $w(o_i^{id})_{csc}$ defines the prior weight of the detected objects based on previous observations (i.e., object detected at $I(k-1)$) as shown in Figure 5.6.

Figure 5.6 presents the process of computing matching weight (w_m) for the observations (i.e., $o_j^{'}$) detected at time k using Equations 5.11, 5.12, and 5.13. In the first, KL divergence is measured among the normalized ellipse histogram of objects detected at k and $k-1$ using Equation 5.12. In the second, CSC-based object prior weight is computed among the color-patches of each object detected at k and $k-1$ using Equation 5.13. These two measured quantities are then combined

[6]We assume that object detected at $I(k-1)$ contains the more probable claims about it next occurrence at $I(k)$.

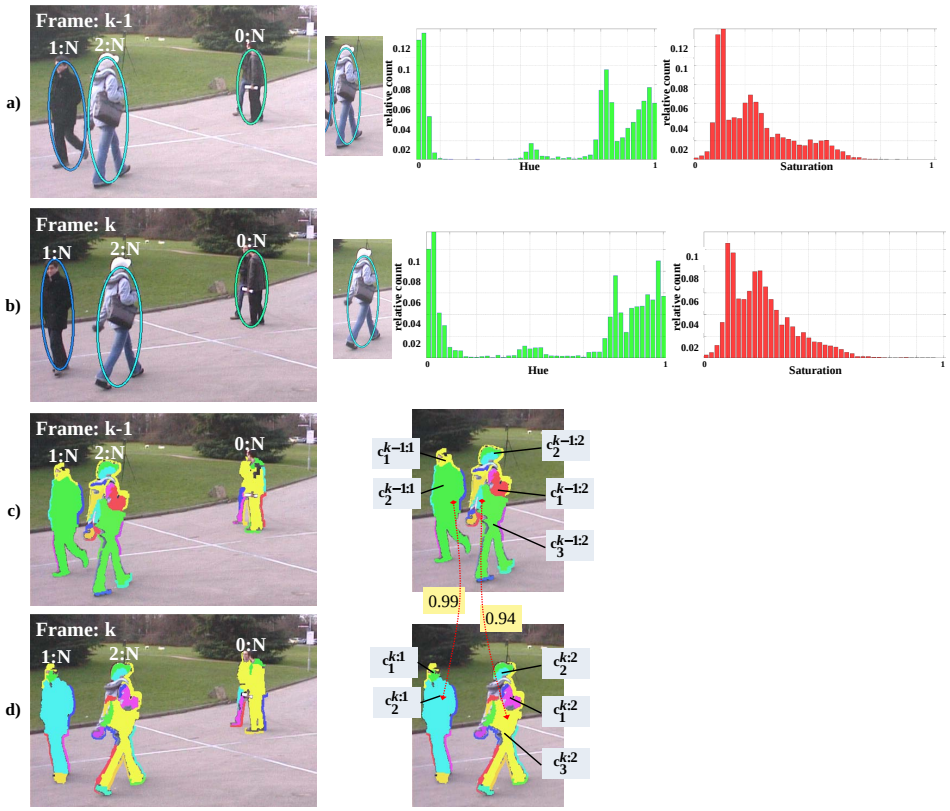


Figure 5.6: shows the computation of matching weights using BMW approach on a sample frame of PETS2009 [3] dataset. a) and b) describe the likelihood measurement process through KL divergence $D(o_j^{'}, o_i^{id})$ among the normalized ellipse histograms of detected objects at frame k and $k-1$. Both normalized Hue (i.e., normalization factor of 360 degrees) and Saturation channel values ranging from 0 to 1 which are divided into 45 intervals. c) and d) show the process of computing prior weight $w(o_i^{id})_{csc}$ by finding the Euclidean distance among the color-patches of objects using CSC approach. The results of matching weight are presented in Table 5.1.

with our BMW (Bayesian Matching Weight) approach in Equation 5.11.

Table 5.1 presents the measured values of KL divergence (i.e., D), CSC-based prior weight (i.e., w_{csc}) and final matching weight (i.e., w_m) of the detected objects at k and $k-1$ in Figure 5.6. In the Table 5.1, KL divergence column shows the measured divergence between the objects detected at k and $k-1$ where the diagonal shows minimum divergences. The CSC-based prior weight column shows the computed prior weight among the color-patches of objects and the diagonal values are averaged to obtain the final prior weight (w_{csc}) for each object. The final matching weight (w_m) column shows the computed matching weight by fusing these

quantities (i.e., D and w_{csc}) using Equation 5.11. However, in ideal situations, these probabilities in Table 5.1 reflect the reliable correspondences among the detected objects in consecutive frames (i.e., k and $k-1$). But in conflicted situations (i.e., occlusion), these measured matching weight become uncertain due to incomplete visual information. Therefore, it is inevitable to look for the alternative ways and to continue the inferring mechanism correctly under conflicted situations. In the following section, we have explained the proposed qualitative approach which work in conjunction with quantitative approach to handle object matching ambiguities under conflicted situations.

Table 5.1: BMW matching approach on a sample frame of PETS2009 [3] dataset in Figure 5.6

KL Divergence			**CSC-based prior-weight**							**BMW (w_m)**		
$\frac{k}{k-1}$	$o_1^{k:1}$	$o_2^{k:2}$	$\frac{k}{k-1}$	$c_1^{k:1}$	$c_2^{k:1}$	$\frac{k}{k-1}$	$c_1^{k:2}$	$c_2^{k:2}$	$c_3^{k:2}$	$\frac{k}{k-1}$	$o_1^{k:1}$	$o_2^{k:2}$
$o_1^{k-1:1}$	0.01	0.72	$c_1^{k-1:1}$	0.99	0.43	$c_1^{k-1:2}$	0.97	0.61	0.63	$o_1^{k-1:1}$	0.96	-
$o_2^{k-1:2}$	0.53	0.05	$c_2^{k-1:1}$	0.52	0.95	$c_2^{k-1:2}$	0.61	0.98	0.31	$o_2^{k-1:2}$	-	0.94
						$c_3^{k-1:2}$	0.28	0.53	0.98			
$exp(-D)$	0.99	0.97	w_{csc}	0.97		w_{csc}	0.97			$P(o_k^{'} = w_m)$	0.96	0.94

5.4.2 Qualitative Approach-Inferencing of Behavioral States

We aimed to incorporate the cognitive model into the vision system to have the similar capabilities as humans for recognizing object activities and behaviors while moving across the scene. So, we can say that the cognitive modeling is empowered by the conceptualization of human perception and inference ways. Therefore, it complements and allows us to interpret the ambiguous (i.e., incomplete or clutter) discrete data adequately in a bidirectional way. To achieve this goal, we have developed axioms inspired from human cognitive abilities. These axioms reliably recognize various states of objects, such as normal walk, occluded, overlaper, etc., and manage identities which dramatically improve the typical tracking mechanism.

In this manner, we are able to acquire the qualitative attributes during motion, such as what is the speed of the object, when the object is occluded, which object is the cause of occlusion and when the object leaves the scene. In addition, we are able to measure the pace and orientation of the objects over time. We have applied Bayesian inference approach to measure the matching weights based on

the visual characteristics of the objects, earlier. The goal of inferring behavioral states during tracking is achieved by the designed axioms. So, the suggested model has a huge impact on behavior, social, and cognitive sciences and we are much closer to develop the intelligent tracking system.

In the following, we have explained the primary concept of modeling human perspective for tracking situations, the fundamentals of developed axioms, and the constraints which are imposed on the inferencing mechanism.

Preliminary Concept with Basic Notations and Definitions

How to handle the uncertainties and ambiguities which are raised from the partial knowledge due to insufficient data or hazy contents? Uncertainties in data reflects the doubt in its source. The uncertainty is usually observed in situations where the actual state of the underlying object is not completely determined, but we have to rely on some human expert's subjective preferences among the different possibilities. In contrast, the notion of ambiguity refers to haziness in data related to a questionable facet of the actual problem. Ambiguity arises whenever a data lacks the desired precision; however, its meaning remains valid. For example, in Figure 5.7(a) there is no ambiguity since only one object is present in the scene but in Figure 5.7(b) and (c), there are multiple objects and there can be uncertainties since the actual outcome (i.e., matching weights from the quantitative approach) is open in terms of accuracy when the identities of the objects are managed.

Before, we describe "the modeling" of axioms for the integrated treatment of uncertainty and ambiguity in the range of knowledge based systems, the description of some concepts and fundamentals are indispensable. According to our understanding, the basic intention of any model is to reflect the properties of the real world such that it enables the prediction of behaviors in the under observed context[7]. Therefore, the model should directly relate itself to the underlying context instead of allowing everything to happen which is of no use. So, the model should be objective and capable of handling the uncertainty and ambiguity. Besides, we have focused on the subjective probability which is widely used in the field of knowledge representation to reflect the degrees of rational belief.

Based on above theoretical aspects, we have investigated the behaviors and properties of moving objects in the world domain and see how a human's cogni-

[7]The term context refers to any particular situation which is taken into account for the experiments and testing purpose.

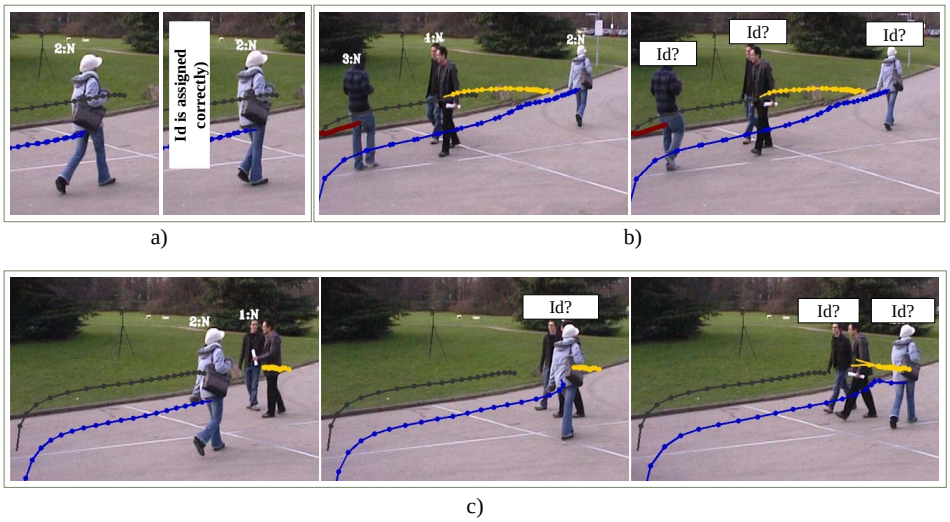


Figure 5.7: shows the examples of potential instances of uncertainty and ambiguity on sample frames of PETS2009 [3] dataset. a) contains only one object in the scene and matching approach performs one-to one matching when object identities are managed, therefore there is no ambiguity in this scene, b) and c) contain multiple objects and there can be uncertainties since the actual outcome (i.e., matching weights from the quantitative approach) is open in terms of accuracy when the identities of the objects are managed.

tive system processes that information along with the quantitative approach. The context of the situation is defined as an observer (i.e., camera) standing at a fixed location and can view a specific range (i.e., a scene). Now, the observer infers and extracts the knowledge from the overall scene where the observations are formulated into the axioms and logical representation, and reasoning models are constructed. Figure 5.3 presents an illustration whereas Table 5.2 shows the observations and their respective inferred knowledge provided by the observer.

It is assumed that the position of each detected object is a continuous function of time in the scene until it leaves permanently. For example, if the object is occluded during overlap, it still exists in its overlper. In our formulation, the time domain is continuous and represented by the discrete real number. In n-dimensional space, the detected objects at k time instance are mapped on the 2D plane as shown in Figure 5.3 whereas their attributes are defined in Equation 5.3. The Q_l demonstrates the behavioral states during tracking which are derived by the developed axioms. The Q_l contains six behavioral states, including normal (N), occluded (Oc), overlaper (Ov), reappear (R), exit (E), and new (Ne) whereas to infer each state, a specific axiom is designed by incorporating the human perception abilities. The developed

axioms contain two types of functions:

- Logical Functions: In axioms, the function take input values and return the logical output (i.e., true or false) whilst satisfying condition. There are three functions performing this task:
  - ◇ $isMax() \Rightarrow return(true|false)$: This function takes the list of objects as inputs along with computed matching weights (i.e., w_m) and returns true if maximum correspondence exists as described in the logical function 1.
  - ◇ $isMin() \Rightarrow return(true|false)$: This function takes the list of objects as inputs along with computed matching weights (i.e., w_m) and returns true if minimum correspondence exists as described in the logical function 2.
  - ◇ $inSearch\_Space() \Rightarrow return(true|false)$: This function checks whether the object is inside the predicted region or not as described in the logical function 3.
- Assignment Functions: In the axioms, these functions perform the task of assignment (e.g., object identity) and update the parameters of the object. There are three functions performing this task:
  - ◇ $Assign\_Id()$: This function assigns the identity to corresponding object.
  - ◇ $Deactive\_Id()$: This function de-activates the object's identity when the object is not present in the scene.

Table 5.2: Inference of object's behavioral states by an observer from a fixed point

Information: Observer	Inferred Behavioral States
object is moving with **normal** pace	normal object (N)
object is entered in the scene and was **not** present earlier	new object (Ne)
object does some functions and **left** the scene	exit object (E)
object during motion **intercepts** the visual appearance of another object	overlaper object (Ov)
object is hidden due to occlusion and **lost** its visual characteristics	occluded object (Oc)
object is reappeared from the interception and **retains** its visibility	reappear object (R)

◇ *Make_Child*(): This function creates a child parent (occluded object as child and overlaper as parent) relationship when occlusion is observed.

Logical Function 1 isMax() Function

if (objects at k finds max correspondence with given observations) **then**
return true
else
return false

Logical Function 2 isMin() Function

if (objects at k finds min correspondence with given observations) **then**
return true
else
return false

Logical Function 3 inSearch_Space() Function

if (object exits any in the predicted region) **then**
return true
else
return false

Tracking Axioms

In this section, the tracking axioms (i.e., abstract qualitative reasoning) are presented. The objects at $I(k)$ and $I(k-1)$ are exploited to infer the behavioral states of the moving object observed at $I(k)$ and to assign the unique identities.

- *Normal State Axiom:* This axiom infers the normal behavioral state of objects based on the assumption that the objects are moving governing the laws of continuous motion with consistent visual attributes.

$$normal(o_j^{id}) = \left\{ isMax_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id}) \wedge \ inSearch\_Space_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id}) \right\}$$

$$Q_l = \{N \rightarrow T, Oc \rightarrow F, Ov \rightarrow F, R \rightarrow F, E \rightarrow F, Ne \rightarrow F\}$$

$$Assign\_Id(o_j^{id}) = \left\{ o_i^{id} \right\}$$

We have developed the above axiom to assign the normal behavioral state $(N \rightarrow T)$ when object is moving with consistent visual appearance. The $normal(o_j^{id})$ state is assigned when two conditions are satisfied. First, the object finds $isMax_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id})$ matching weight with its previous possibilities

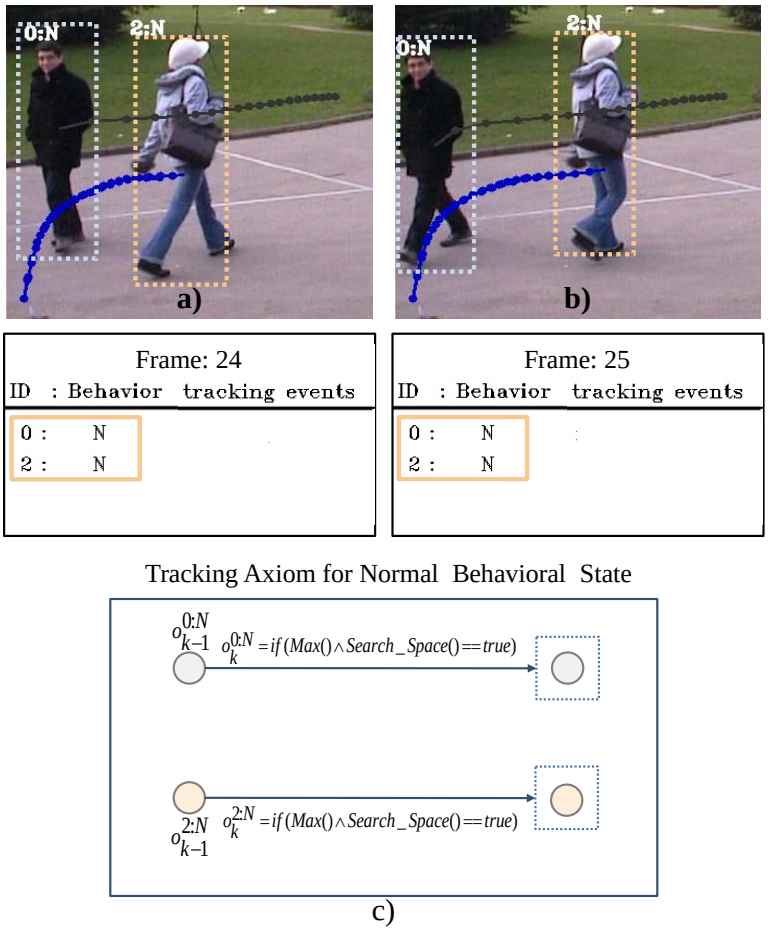


Figure 5.8: shows the logical inferencing for normal behavioral state on sample frame of PETS2009 [3] dataset. a) and b) show the visual representation of two consecutive frames where two objects are moving with normal behaviors N and unique identities 0 and 2. c) shows the logical interpretation of behavior inference mechanism which is based on two conditions: first, object finds $isMax()$ relationship (i.e., matching weight) with its previous possibilities at frame $k-1$; second, object should fall in $inSearch\_Space()$ region predicted based on previous possibilities at frame $k-1$. When these conditions are satisfied (i.e., $true$), the $normal(o_j^{id})$ behavioral state along with respective identity is assigned to objects.

o_i^{id} at frame $k-1$. Second, object should falls in $inSearch\_Space_{o_j' \in I(k)}(o_j', o_i^{id})$ region which is predicted based on previous possibilities o_i^{id} at frame $k-1$. When these two conditions are satisfied, the $normal(o_j^{id})$ behavioral state is assigned to objects. The qualitative list of behaviors Q_l is updated where normal behavioral state is set to true[8] ($N \rightarrow T$) and other behaviors are assi-

[8]In all of these axioms, T refers to true and F refers to false flag as short terms representation.

gned false flag. In the last, the identity of the previous possibilities is assigned to the detected object at frame k through $Assign\_Id(o_j^{id})$ function. Figure 5.8(a-b) demonstrates this situation on sample frame of PETS2009 [3] dataset where two objects are moving with normal conditions while maintaining their identities in consecutive frames. Figure 5.8(c) describes the logical interpretation of $normal(o_j^{id})$ axiom.

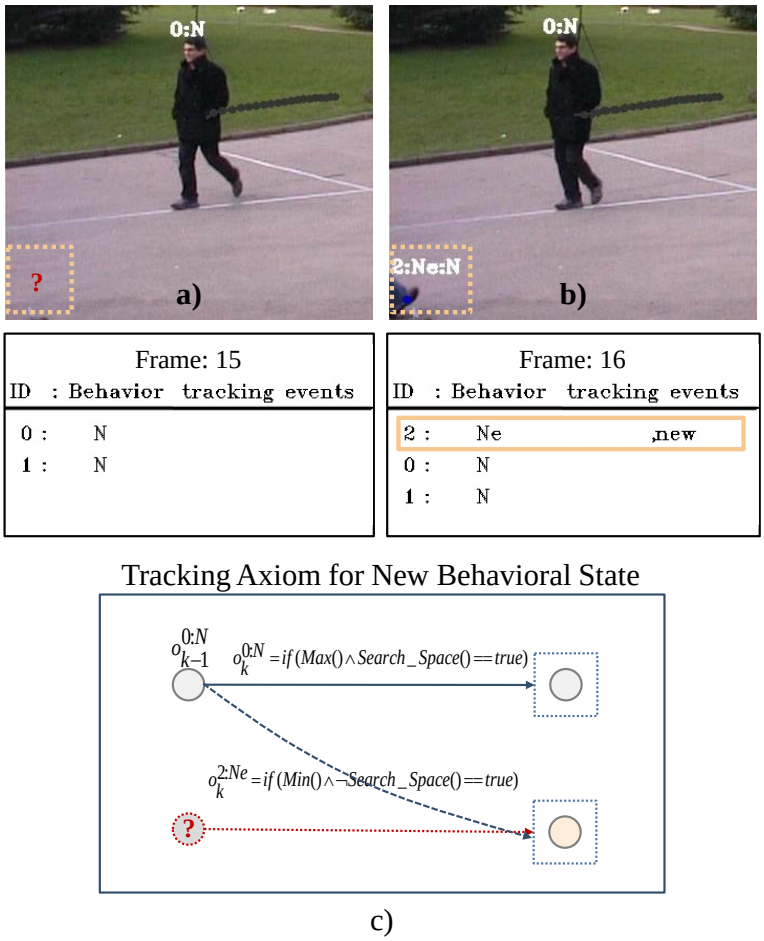


Figure 5.9: shows the logical inferencing for new behavioral state on sample frame of PETS2009 [3] dataset. a) and b) show the visual representations of two consecutive frames where a new object is entered in the scene and assigned new behavioral state (Ne) along with id *2*. c) shows the logical interpretation of behavior inference mechanism based on two conditions: first, object finds $isMin()$ relationship with previous possibilities at $k-1$; second, object does not fall in any predicted $inSearch\_Space()$ region at Frame $k-1$. When these conditions are satisfied (i.e., $true$), $new(o_j^{'})$ behavioral state with new id is assigned to object.

- *New State Axiom:* This axiom infers the new behavioral state of the object and is triggered when new tracking event n_{active} occurs.

$$if\ n_{active} == true$$
$$new(o_j^{'}) = \left\{ isMin_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id}) \wedge \neg\ inSearch\_Space_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id}) \right\}$$
$$Q_l = \{N \rightarrow T, Oc \rightarrow F, Ov \rightarrow F, R \rightarrow F, E \rightarrow F, Ne \rightarrow T\}$$
$$Assign\_Id(o_j^{id}) = \{id_{new}\}$$

We have developed the above axiom to assign the new behavioral state $(Ne \rightarrow T)$ when a new object is entered in the scene. The $new(o_j^{'})$ state is assigned when two conditions are satisfied. First, the object finds $isMin_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id})$ correspondence (i.e., matching weight) with the previous possibilities o_i^{id} at frame $k-1$. Second, object does not fall in any $inSearch\_Space_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id})$ region which is predicted based on previous possibilities o_i^{id} at frame $k-1$. When these two conditions are satisfied, the $new(o_j^{'})$ behavioral state is assigned to object. The qualitative list of behaviors Q_l is updated where both new and normal behavioral states are set to true $(N \rightarrow T, Ne \rightarrow T)$ and the other behaviors are assigned false flag. In the last, a new unique identity (id_{new}) is assigned to the object from the identity pool. In Figure 5.9(a-b), we have demonstrated this situation on sample frame of PETS2009 [3] dataset when an object is entered in the *Frame 16*. Figure 5.9(c) describes the logical interpretation of $new(o_j^{'})$ axiom.

- *Exit State Axiom:* This axiom assigns the exit behavioral state when the object leaves the scene (i.e., field of view).

$$if\ e_{exit} == true$$
$$exit(o_i^{id}) = \left\{ isMin_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id}) \wedge \neg\ inSearch\_Space_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id}) \right\}$$
$$Q_l = \{N \rightarrow F, Oc \rightarrow F, Ov \rightarrow F, R \rightarrow F, E \rightarrow T, Ne \rightarrow F\}$$
$$Deative\_Id(o_i^{id})_{o_i^{'} \in I(k-1)} = \left\{ o_i^{id} \right\}$$

We have developed the above axiom to assign the exit behavioral state $(E \rightarrow T)$ when the object leaves the scene's field of view. The $exit(o_i^{id})$ state is assigned when two conditions are satisfied. First, object finds $isMin_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id})$

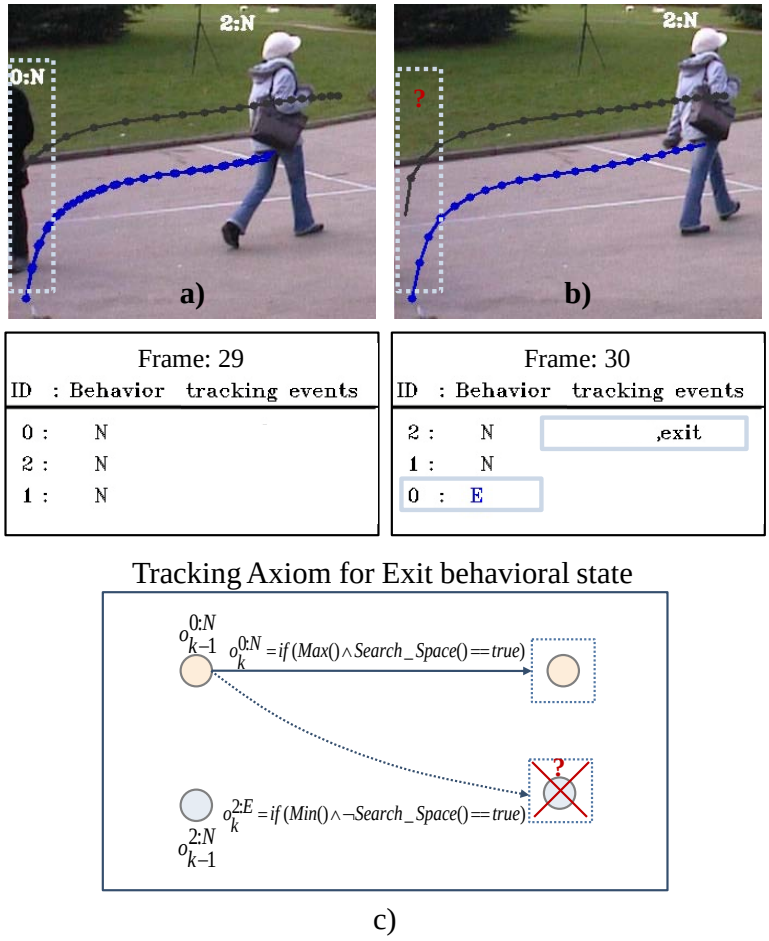


Figure 5.10: shows the logical inferencing for exit behavioral state on sample frame of PETS2009 [3] dataset. a) and b) show the visual representations of two consecutive frames where the object with id 0 left the scene in *Frame 30* and assigned exit behavioral state (E). c) shows the logical interpretation of behavior inference mechanism which is based on two conditions: first, the object finds $isMin()$ relationship (i.e., matching weight) with possibilities at frame k; second, the object does not fall in any $inSearch\_Space()$ region. When these two conditions are satisfied (i.e., $true$), the $exit(o_i^{id})$ behavioral state is assigned along with deactivating the unique identity of the object.

correspondence (i.e., matching weight) with the possibilities o_i^{id} at frame $k-1$. Second, object does not fall in any $inSearch\_Space_{o'_j \in I(k)}(o'_j, o_i^{id})$ region. When these two conditions are satisfied, $exit(o_i^{id})$ behavioral state is assigned to that object. The qualitative list of behaviors Q_l is updated where exit behavioral state is set to true ($E \rightarrow T$) and the other behaviors are assigned false flag. In the last, the assigned unique identity is deactiva-

ted $Deative\_Id(o_i^{id})_{o_i^{'}\in I(k-1)}$. Figure 5.10(a-b), demonstrates this situation on sample frame of PETS2009 [3] dataset at *Frame 30*. Figure 5.10(c) describes the logical interpretation of $exit(o_i^{id})$ axiom.

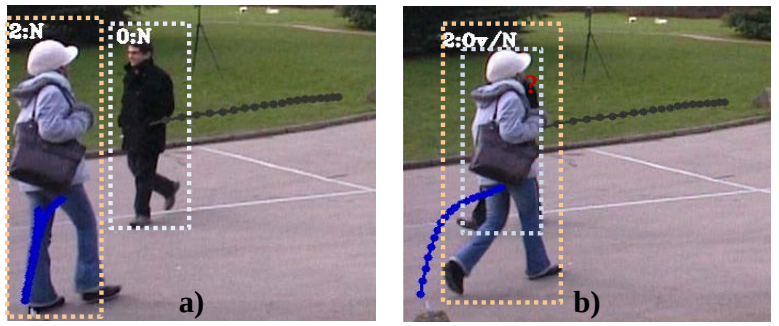
- *Overlaper State Axiom:* This axiom infers the overlaper behavioral state when occlusion tracking event o_{active} is activated.

$$if\ o_{active} == true$$
$$overlaper(o_j^{id}) = \left\{ isMax_{o_j^{'}\in I(k)}(o_j^{'}, o_i^{id}) \wedge\ inSearch\_Space_{o_j^{'}\in I(k)}(o_j^{'}, o_i^{id}) \right\}$$
$$Q_l = \{N \rightarrow T, Oc \rightarrow F, Ov \rightarrow T, R \rightarrow F, E \rightarrow F, Ne \rightarrow F\}$$
$$Assign\_Id(o_j^{id}) = \left\{ o_i^{id} \right\}$$
$$Make\_Child(o^{id}_{j+1\ if(\exists(o^{id}_{j+1}\in I(k))\in(Q_l=\{oc==T\}))}, o^{id}_{j\ if(\exists(o^{id}_{j}\in I(k))\in(Q_l=\{ov==T\}))})$$

The axiom is developed to assign the overlaper behavioral state $(Ov \rightarrow T)$ when the objects intercept each other during motion. The $overlaper(o_j^{id})$ state is assigned when two conditions are satisfied. First, object finds the $isMax_{o_j^{'}\in I(k)}(o_j^{'}, o_i^{id})$ correspondence (i.e., matching weight) with the possibilities o_i^{id} at frame $k-1$. Second, object falls in $inSearch\_Space_{o_j^{'}\in I(k)}(o_j^{'}, o_i^{id})$ region. When these two conditions are satisfied, $overlaper(o_j^{id})$ behavioral state is assigned to object. The qualitative list of behaviors Q_l is updated where both overlaper and normal behavioral state[9] are set to true $(Ov \rightarrow T, N \rightarrow T)$ and the other behaviors are assigned false flag. The unique identity is transferred with $Assign\_Id(o_j^{id})$ function. $Make\_Child()$ function creates a parent-child relationship, and overlaper becomes the parent of the occluded object. After the occlusion, child adopts the visual features of its parent and is updated frame-by-frame using depth first search strategy. Figure 5.11(a-b) demonstrates this situation on sample frame of PETS2009 [3] dataset when two objects occlude each other in *Frame 20*. Figure 5.11(c) describes the logical interpretation of $overlaper(o_j^{id})$ axiom.

- *Occluded State Axiom:* This axiom infers the occluded behavioral state when object disappears due to the interception with other object.

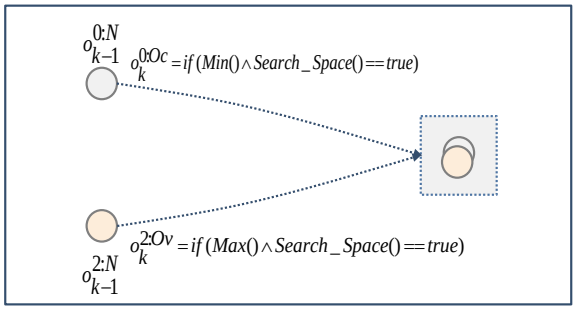
[9] As, the overlaper object keeps its visual attributes while occlusion, therefore, we have also assigned the normal behavior along with overlaper behavior.



Frame: 19		
ID	: Behavior	tracking events
0 :	N	
2 :	N	
1 :	N	

Frame: 20		
ID	: Behavior	tracking events
2 :	Ov/N	,occlusio
1 :	N	
0 :	Oc	

Tracking Axiom for Occluded and Overlaper Object



c)

Figure 5.11: shows the logical inferencing for occluded and overlaper behavioral states during occlusion on sample frame of PETS2009 [3] dataset. a) and b) show the visual representations of two consecutive frames where objects with identities 0 and 2 intercept each other in *Frame 20*. It is notable that object with identity 2 keeps its visual appearance visible and hides the object with identity 0, so it is assigned overlaper behavioral state *Ov*. In contrast, object with identity 0 is fully occluded and assigned occluded behavioral state *Oc*. c) shows the logical interpretation of behavior inference mechanism which is based on two conditions: first, measuring the objects relationship with possibilities at frame $k-1$; second, searching objects the predicted in $inSearch\_Space()$ region. When these two conditions are satisfied the corresponding behavioral states are assigned to objects.

$$if\ o_{active} == true$$

$$occluded(o_j^{id}) = \left\{ isMin_{o_j' \in I(k)}(o_j', o_i^{id}) \wedge\ inSearch\_Space_{o_j' \in I(k)}(o_j', o_i^{id}) \right\}$$

$$Q_l = \{N \rightarrow T, Oc \rightarrow T, Ov \rightarrow F, R \rightarrow F, E \rightarrow F, Ne \rightarrow F\}$$

$$Assign\_Id(o_j^{id}) = \left\{ o_i^{id} \right\}$$

We have developed the above axiom to assign the occluded behavioral state $(Oc \rightarrow T)$ when objects intercept each other during motion. The $occluded(o_j^{id})$ state is assigned when two conditions are satisfied. First, object finds the $isMin_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id})$ correspondence (i.e., matching weight) with the possibilities o_i^{id} at $k-1$. Second, the object falls in the $inSearch\_Space_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{id})$ region. When these two conditions are satisfied, the $occluded(o_j^{id})$ behavioral state is assigned to object and it becomes the child of the overlaper object. The qualitative list of behaviors Q_l is updated where occluded behavioral state is set to true $(Oc \rightarrow T)$ and the other behaviors are assigned false flag. The unique identity is transferred with $Assign\_Id(o_j^{id})$ function. Figure 5.11(a-b) demonstrates this situation on sample frame of PETS2009 [3] dataset when two objects occlude each other in *Frame 20*. In Figure 5.11(c), describes the logical interpretation of the $occluded(o_j^{id})$ axiom.

- *Reappear State Axiom:* This axiom infers the object's reappear behavioral state when occlusion is ended and split tracking event s_{active} is activated.

$$
\begin{aligned}
&if\ s_{active} == true \\
&reappear(o_j^{id}) = \left\{ isMax_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{*id}) \right\} \\
&Q_l = \{N \rightarrow T, Oc \rightarrow F, Ov \rightarrow F, R \rightarrow T, E \rightarrow F, Ne \rightarrow F\} \\
&Assign\_Id(o_j^{id}) = \left\{ o_i^{*id} \right\}
\end{aligned}
$$

The above axiom is developed to assign the reappear behavioral state $(R \rightarrow T)$ when visual interception among the objects is ended. The $reappear(o_j^{id})$ state is assigned when object finds $isMax_{o_j^{'} \in I(k)}(o_j^{'}, o_i^{*id})$ correspondence (i.e., matching weight) with the list of occluded objects o_i^{*id}. When this condition is satisfied $reappear(o_j^{id})$ behavioral state is assigned to that object. The qualitative list of behaviors Q_l is updated where both reappeared and normal behavioral states are set to true $(R \rightarrow T, N \rightarrow T)$ and the other behaviors are assigned false flag. The unique identity of occluded object is assigned with $Assign\_Id(o_j^{id})$ function. Figure 5.12(a-b) demonstrates this situation on sample frame of PETS2009 [3] dataset when the object is reappeared in *Frame 23*. Figure 5.12(c) describes the logical interpretation of $reappear(o_j^{id})$ axiom.

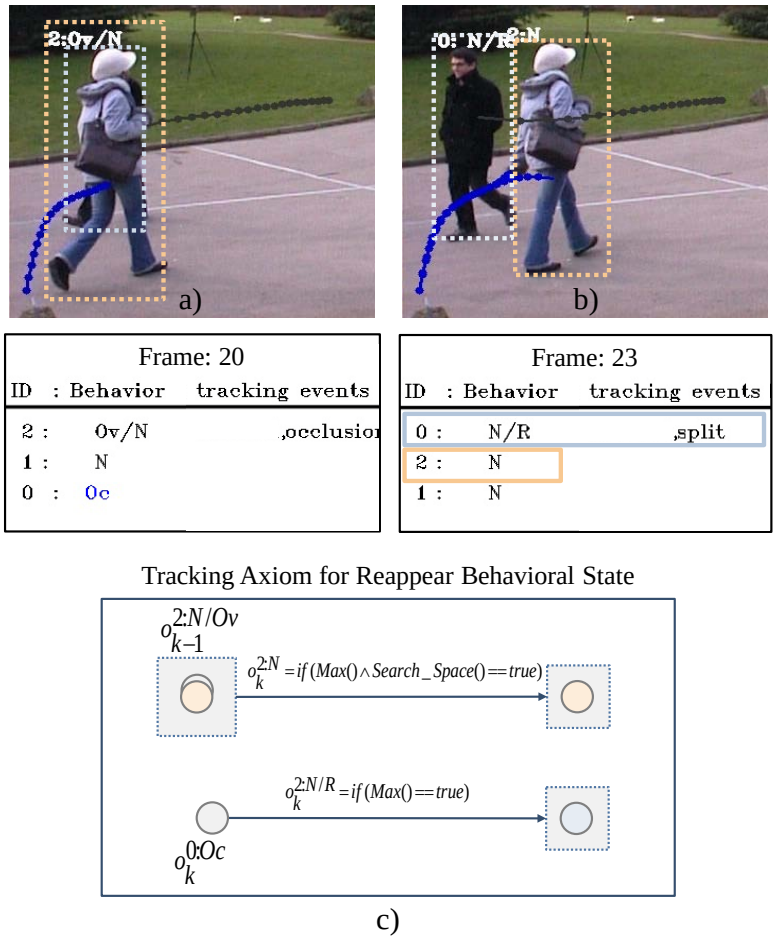


Figure 5.12: shows the logical inferencing for reappear behavioral state a on sample frame of PETS2009 [3] dataset. a) and b) show the visual representation of two consecutive frames where object with identity 0 is reappeared (R) from occluded state in *Frame 23*. c) shows the logical interpretation of behavior inference mechanism which is based on a condition that object finds maximum correspondence (i.e., matching weight) with list of occluded objects. When this condition is satisfied (i.e., $true$), the corresponding behavioral state (i.e., reappear) is assigned to object.

Integrity Constraints

In this section, we have defined the constraints which are modeled by keeping object behaviors under consideration. These constraints work with tracking axioms to incorporate the real-time situations and motion-based obligations.

- *Constraint 1:* when the process of tracking is initialized, $k = 1$, all the objects detected at $I(k)$ are labelled as new ($Ne \rightarrow T$) and normal ($N \rightarrow T$) behavioral

states. This statement is only valid for initialization process.

$$\forall Q_l = \{N \rightarrow T \wedge Ne \rightarrow T\} \quad if\ k = 1$$

- *Constraint 2:* no behavioral state is assigned other than normal behavior (N) during ideal tracking. If any other state is activated, then it should be discarded until the valid tracking event is activated.

$$\exists Q_l = \{N \rightarrow T\}$$

- *Constraint 3:* when the occlusion event $o_{active} = true$ is observed, the object which is assigned the overlaper state ($Ov \rightarrow T$) also possess the normal state ($N \rightarrow T$). During occlusion, the visual characteristics are not intruded and the association is possible over time unlike occluded object.

$$\exists Q_l = \{N \rightarrow T \wedge Ov \rightarrow T\}$$

- *Constraint 4:* when an object is overlapped by another object, then its behavioral states is set to occluded ($Oc \rightarrow T$) and all the other behaviors are assigned false flags (i.e., F).

$$\exists Q_l (Oc \rightarrow T) \rightarrow ((N \wedge Ne \wedge R \wedge E \wedge Ov) \rightarrow F))$$

In this approach, each detected object at $I(k)$ is assigned a unique id with respective behavioral states[10]. Now, the next task is to perform object localization at each time instance. In the following, we have developed a Kalman filter based tracking system to estimate the object trajectories over time.

5.5 Kalman Filter Based Tracking System

One of the essential steps in tracking framework is to localize the objects. To achieve this task, many algorithms have been proposed that are used for stochastic estimation of object locations (i.e., trajectory) from noisy measurements for instance, object location and its velocity. One of the well-known approaches is

[10]Normal (N), New (Ne), Exit (E), Overlaper (Ov), Occluded (Oc), Reappear (R).

Kalman filter [109], a recursive solution to the discrete data linear filtering problem. However, originally this tool is used for linear systems while implying Kalman filter in tracking problem may violate the linearity condition. To address this fundamental limitation, various modifications have been suggested in literature for instance, Extended Kalman filter. However, in our suggested approach, the linearity condition is maintained even under non-linear situations. So, we have extended the applicability of original Kalman filter for both the linear and non-linear system without making any modification in its actual contents. Practically, the basics of the Kalman filter are a set of mathematical equations that operates recursively in the predictor-corrector way to minimize the estimated error covariance until some presumed conditions are met [110]. Kalman filter has been an extensively researched topic, particularly in the area of assisted navigation and object tracking in computer vision.

In our tracking framework, each detected object with unique identity is modeled as a linear system and the Kalman filter is used to estimate the state of the objects at each time instance k. For this purpose, the available measurements z_k are exploited to estimate[11] the state of the object $\vec{x}_k$. Before, starting the estimation process, the estimator should take these requirements into account after satisfying certain assumptions about the noise that affects the tracking system of an object:

- the average value of our state estimate is equal to the average value of the true state to avoid biased state estimation. Mathematically, the expected value of the estimate should be equal to the expected value of the state.

- a state estimate should deviate from the true state as low as possible. Mathematically, the estimator should find the smallest possible error variance.

The Kalman filter based tracking system for each detected object employed here is derived from its original formulation where the measurements $z \in \mathcal{R}^m$ and state of the object $x \in \mathcal{R}^n$ are observed at each time instance k. In our tracking system, each Kalman filter is described by its process $\vec{x}_k$ which is defined by the center of gravity (i.e., x_s and y_s) and speed of objects (i.e., dx_s/dk and dy_s/dk), measurement model $\vec{z}_k$, and available information about the model's initial conditions which are

[11] As, the objects localized (i.e., system) behavior can be expected according to the state equation given the measurement information of the position and velocities so, how can we determine the best estimate of the state $\vec{x}_k$?

governed by linear stochastic difference and measurement equation respectively:

$$\vec{x_k} = \begin{bmatrix} x_s \\ y_s \\ dx_s/dk \\ dy_s/dk \end{bmatrix}, \quad \vec{z_k} = \begin{bmatrix} x_m \\ y_m \\ dx_m/dk \\ dy_m/dk \end{bmatrix}, \tag{5.14}$$

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \tag{5.15}$$

$$\vec{x_k} = A\vec{x}_{k-1} + \vec{w}_{k-1}, \tag{5.16}$$

$$\vec{z_k} = H\vec{x_k} + \vec{v_k}, \tag{5.17}$$

The matrix A in Equation 5.16 relates the state at the previous time $k-1$ to the state at current time k, in the absence of either a driving function or process noise $\vec{w}_{k-1}$. H in the measurement Equation 5.17 relates the state $\vec{x_k}$ to the measurement $\vec{z_k}$. In practice, both A and H matrices can change with each time step, but here we assume that it is constant.

The random variables $\vec{w}_{k-1}$ and $\vec{v_k}$ in Equations 5.16 and 5.17 represent the process and measurement Gaussian white noise, respectively. Both noises are independent of each other and are defined as:

$$P(w) \approx \mathcal{N}(0, Q) \tag{5.18}$$

$$P(v) \approx \mathcal{N}(0, R) \tag{5.19}$$

In practice, the process noise covariance Q and measurement noise covariance R matrices might change with each time step or measurement, but we assume that it remains constant.

A Kalman filter based tracker estimates a process using a form of feedback control: the filter estimates the process state at some time k and then obtains the feedback in the form of noisy measurements. The equations for the Kalman filter fall into two groups: time-update equations and measurement-update equations. The time-update equations are responsible for projecting forward (i.e., in time), the current state and error covariance estimates to obtain the a priori estimates for the next time step. The measurement-update equations are responsible for the

feedback (i.e., for incorporating a new measurement into a priori estimate) to obtain an improved a posteriori estimate. Moreover, the time-update equations can also be thought of as predictor equations, while the measurement-update equations can be thought of as corrector equations. Indeed, the final estimation algorithm resembles that of a predictor-corrector algorithm for solving the numerical problems. Both time and measurement update are presented as follows:

5.5.1 Time-Update Equations

In the time-update equation, the process begins with system initialization by assuming the initial values for error covariance estimate with P_{k-1} *or* $P_0 = 0$. So the filter begins with an initial a posteriori state estimate $\hat{x}_k$ *or* $\hat{x}_0 = 0$. The time-update equation [110] are:

$$\vec{x}_k^- = A\vec{x}_{k-1} \tag{5.20}$$

$$P_k^- = AP_{k-1}A^T + Q \tag{5.21}$$

where the time-update Equations 5.20, and 5.21 project the state and covariance estimates forward from time $k-1$ to k where Q is from Equation 5.18.

5.5.2 Measurement-Update Equations

The measurement-update [110] Equations 5.22, 5.23, and 5.24 are defined as:

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \tag{5.22}$$

$$\vec{x}_k = \vec{x}_k^- + K_k(\vec{z}_k - H\vec{x}_k^-) \tag{5.23}$$

$$P_k = (I - K_k H)P_k^- \tag{5.24}$$

The first task during the measurement-update is to compute the Kalman gain K_k in Equation 5.22. The next step is to measure the actual process to obtain $\vec{z}_k$, and then to generate a posteriori state estimate $\vec{x}_k$ by incorporating the measurement in Equation 5.23. The final step is to obtain a posteriori error covariance estimate P_k from Equation 5.24. After each time and measurement-update pair, the process is repeated with the previous a posteriori estimate that is used to project or predict new a priori estimates.

In our framework shown in Figure 5.1, when a new moving object is detected, a new tracker is assigned to it which estimates the object locations based on given

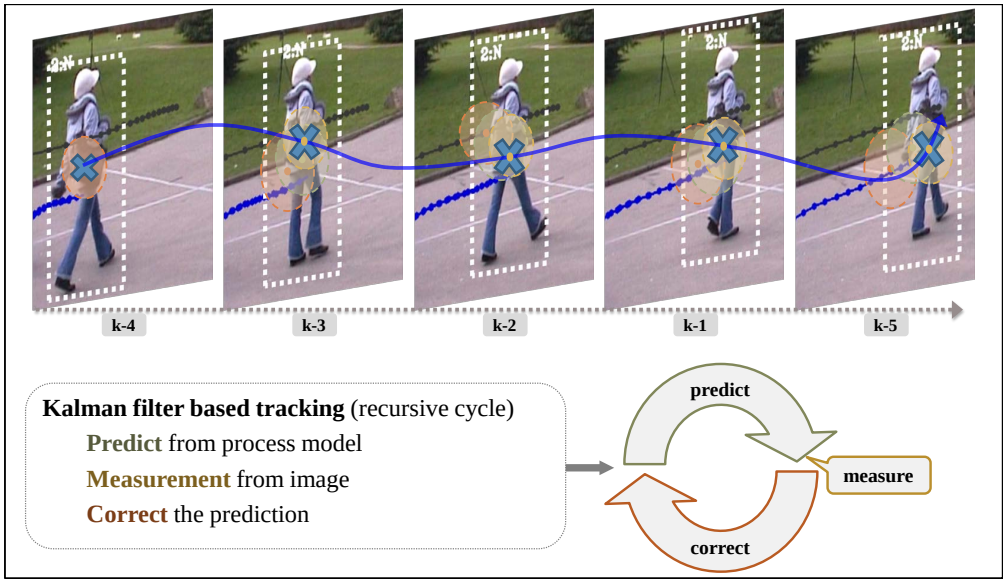


Figure 5.13: shows the object localization process of Kalman filter based tracking on a sample frame of PETS2009 [3] sequence. During object localization, a Kalman filter based tracker estimates a process using a form of feedback control: the filter estimates (i.e., predict) the process state at some time k and then obtains the feedback in the form of noisy measurements (i.e., correct).

states $\vec{x_k}$ and measurement $\vec{z_k}$ values. After the initialization, in the next frames, the normal state updation continues until any other tracking events (i.e., occlusion, split, new, or exit) are detected which is a linear situation for Kalman filter based tracker as shown in Figure 5.13. However, when the objects are occluded, the linearity of the tracking system is effected. In this non-linear situation, the Kalman filter of the occluded object follows the states and measurement information of the corresponding overlaper object. In contrast, during the split, the Kalman filter resumes the tracking by taking into account the parameters of its own object to perform estimations. In this manner, we are able to perform object localization with a classical Kalman filter in a linear and non-linear situations under occlusion and split of objects in complex scenarios.

5.6 Experiments and Discussion

In this section, we demonstrate the performance of proposed quantitative and qualitative approaches in the tracking framework. Several experiments (see Appendix A.4 for more results) are conducted on benchmark video sequences. Besides,

Section 5.6.4 and 5.6.5 provide a detailed quantitative analysis along with discussion of results and context of applicability.

5.6.1 IESK Dataset

The initial dataset is selected for the development of ideas intentionality from IESK, OvG University called IESK dataset. The videos are filmed in the vicinity of campus to capture the video footages containing the real attitude of objects using a single static camera. The IESK dataset been chosen for the following demonstration because it has many potential difficulties:

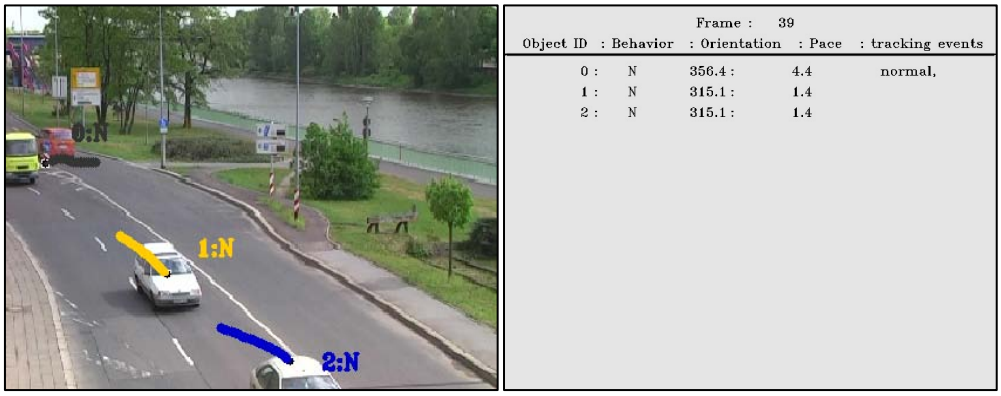
- no prior arrangement is done to avoid any fabricated situations,
- scene noise particularly camera jitter due to varying and windy weather,
- high shadows are observed due to weather variations,
- objects are occluded multiple times during motion over time.

Figures 5.14, 5.15, 5.16 and 5.17 show key frames from important instances of the sequences. The results are visualized by trajectory of the object and labeling of object identity with respective behavior. Moreover, Tracking and Behavior Information Interface (TnBII) Panel is the program interface which describes the overall information about all the information that includes: object identities with behaviors, object pace, orientation, and tracking events for each corresponding frame. Figure 5.14 demonstrates the beginning of tracking and behavior understanding task as:

Frame 39: This frame shows the initial situation of the tracking and behavior understanding. All the detected objects are assigned unique ids *0*, *1*, and *2* with trajectories indicating their tracks. The *TnBII* panel demonstrates the pace (i.e., pixel motion) and orientation (direction in degree) of the objects. It is observed that yellow truck and red cars shared similar spatial region. Therefore, it is not possible to distinguish between them and the same identity is assigned.

Figure 5.15(a) and (b) demonstrate the tracking in complex situations as:

Frame 67: In this frame, the occlusion is observed multiple times for the object with id *3* during parallel and cross movements in the scene with objects having ids *0* and *4*. Moreover, the tracking event indicates that occlusion event



a) b)

Figure 5.14: shows the results on IESK dataset for *Frame 39*. a) shows the visual scene captured for *Frame 39* which represents the ideal situation where three objects are detected and tracked. b) shows the program interface which is named as *TnBII* panel to demonstrate information about behavior, orientation (i.e., degrees), and pace (i.e., pixel motion).

is observed in *TnBII* panel, and it is demonstrated that object with id *3* is occluded by the object with id *0*. It is also observed that pace of the object with id *0* is increased due to occlusion.

Frame 76: This frame shows the split event and its impact on objects trajectories and behaviors. Moreover, a new object is appeared in the scene and assigned a unique id *5* from the identity pool. The object with id *3* carries the reappear behavior and continues its motion in the scene. The *TnBII* panel demonstrates the adapted information, such as the objects with ids *1* and *2* exit from the scene, the objects with ids *0* and *4* are moving with normal behavior.

Figure 5.16(a) and (b) demonstrate the tracking during occlusion and split situations as:

Frame 82: This frame shows the occlusion situation which is observed between objects with id *0* and *5*. The object with id *0* is set to overlaper behavioral state whereas the object with id *5* indicates the occluded behavior. It is also noticed that object with id *4* and *3* is also under occlusion and the respective behaviors are indicated in *TnBII* panel.

Frame 88: This frame shows the split event and its impact on objects trajectories and behaviors. The occlusion between objects with id *0* and *5* is ended, and the object with id *5* is reappeared and continues its normal motion. The *TnBII*

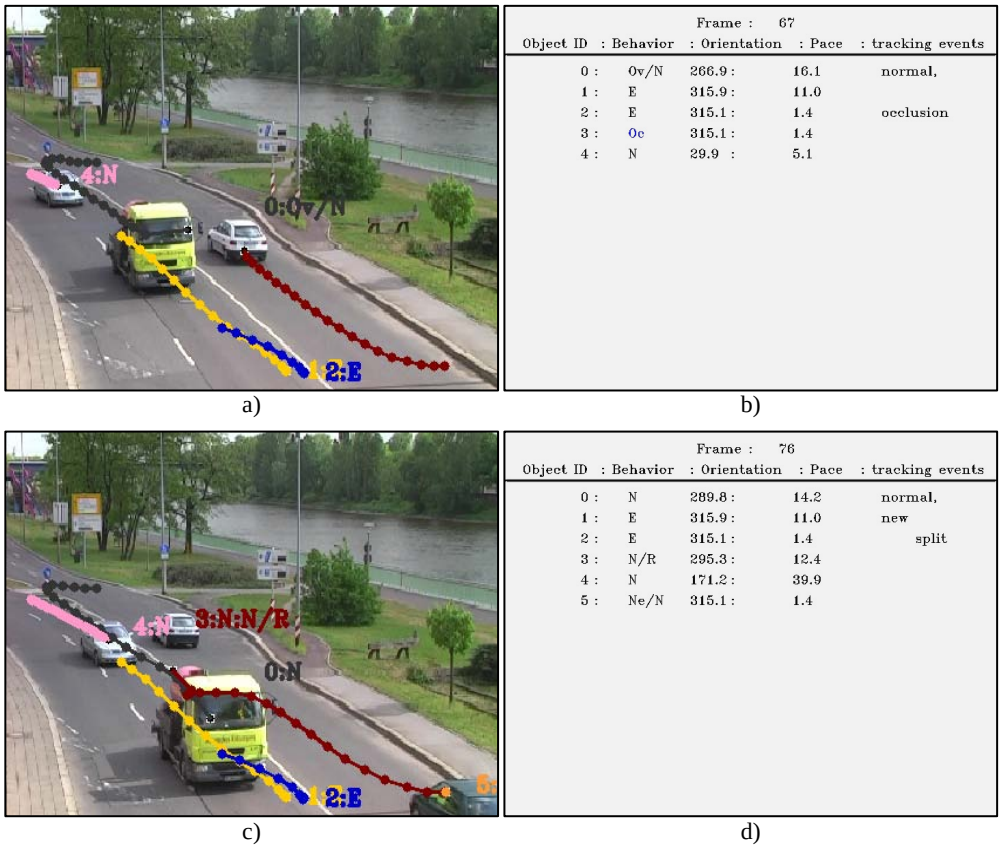


Figure 5.15: shows the results of tracking and behavior understanding on IESK dataset for *Frame 67* and *Frame 76*. a) and c) show the visual scene captured for *Frame 67* and *Frame 76* which are representing the occlusion and split situation of objects with ids *0* and *3*. In b) and d), the *TnBII* panel demonstrates the behavioral state of the objects along with orientation and pace. It can also noticed that both occlusion and split events are active in corresponding frames.

panel demonstrates the updated information about the object behaviors in the scene.

Figure 5.17(a) and (b) demonstrate specifically the new and exit tracking events with corresponding behaviors as:

Frame 49: This frame shows a new object which is entered in the scene from the back view and is assigned a unique id *3*. This object continues its path till the end while the events of occlusion and split are observed multiple times, and the behavior of the object is updated accordingly.

Frame 103: In this frame, object with id *3* has left the scene. Its respective tracker terminates the estimation task, and the identity is released so that it will

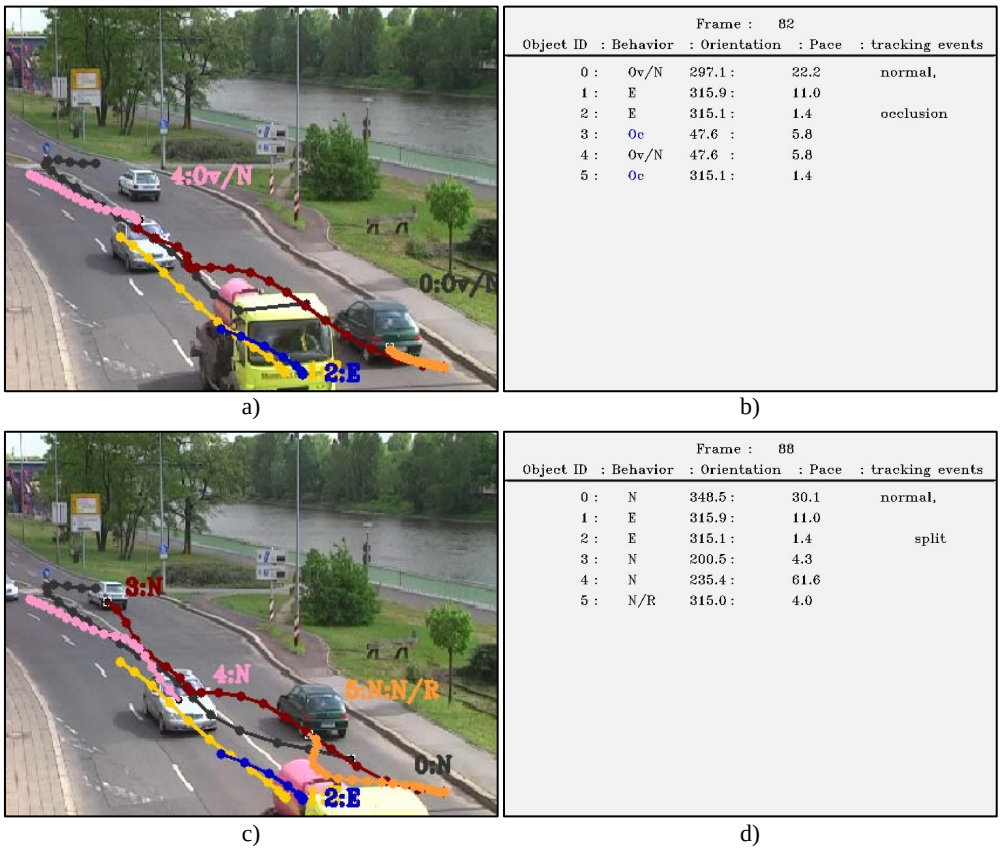


Figure 5.16: shows the results of tracking and behavior understanding on IESK dataset for *Frame 82* and *Frame 88*. a) and c) show the visual scene captured for *Frame 82* and *Frame 88* which are representing the occlusion and split situation for objects with id *0* and *5*. In b) and d), *TnBII* panel demonstrates behaviors state of the objects along with orientation and pace. It can also be noticed that both occlusion and split events are active in corresponding frames.

be assigned to other objects. It is notable from the trajectory of the object shows that how the object keeps its track during motion. The *TnBII* panel demonstrates updated information about the object behavior, its pace, and orientation along with the respective tracking events.

Discussion

The proposed approach is tested on a traffic sequence where objects are moving in both crossing and parallel tracks as shown in above Figures. This complexity is due to the frequent occlusions and separations which are observed in short time intervals. Moreover, camera is not facing the road instead it is tilted which results

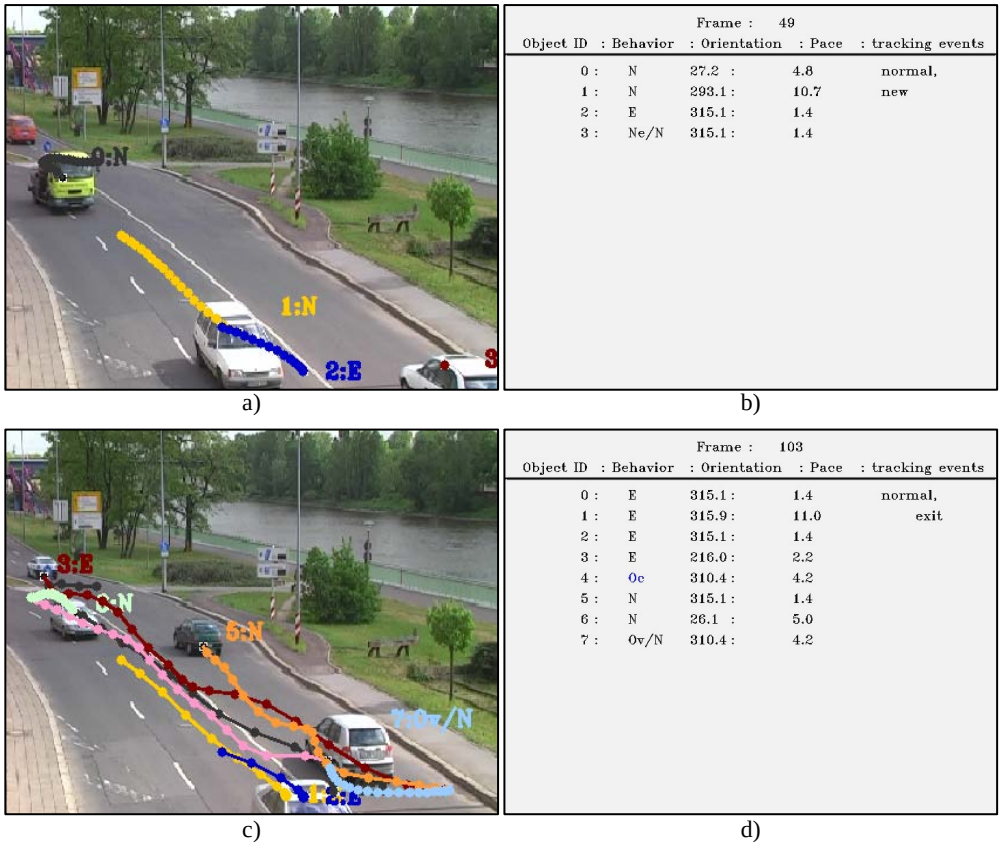


Figure 5.17: shows the results of tracking and behavior understanding on IESK dataset for *Frame 49* and *Frame 103*. a) and c) show the visual scene captured for *Frame 49* and *Frame 103* which are representing the entry and exit situations of object id *3*. In b) and d), *TnBII* panel demonstrates the behavioral state of the objects along with orientation and pace. It can also noticed from the tracks that this object has undergone many occlusions and splits while crossing through the scene.

in a perspective view. Due to this fact, the significant variation in object's size and proximity is observed from start to end. Another challenging aspect of this sequence is the variation in lights due to the weather (i.e., cloudy to sunny). Therefore, it is improved with our object detection method. Throughout the scene, all real-time events are observed for instance, new, exit, occlusion, and split.

5.6.2 PETS2006 Dataset

The second dataset which is used for testing is taken from PETS2006 [1]. The PETS series of workshops make available public datasets for tracking and behavior understanding tasks which are provided with information about the actions,

a) b)

Figure 5.18: shows the results of tracking and behavior understanding on PETS2006 [1] dataset for *Frame 3*. a) shows visual scene which is captured for *Frame 3* representing the ideal situation where three objects are detected and tracked. b) *TnBII* panel demonstrates information about behavioral states, orientation, and pace of the object.

locations and behaviors of the actors contained in it. Moreover, the PETS2006 [1] video sequence has been chosen for the following demonstration because it has many potential difficulties for people tracking:

- people strolling in the scene on the usual walk way,
- people occlude each other as they walk,
- high shadows of people due to strong background reflections,
- multiple people occlude each other and parted again.

Figure 5.18 demonstrates the tracking in normal situation as:

Frame 3: The detected objects are identified by their unique identities and associated behavioral states from our integrated approach of the framework where the corresponding trajectories demonstrate the outcome of tracker. In the following, it is examined that how the proposed approach functions in complex situations.

Figure 5.19(a) and (b) show the complex situations:

Frame 22: This scene shows the occlusion situation when multiple objects with ids *1*, *2*, and *3* interact with each other. As a consequence, the object's tracker is unable to advance the localization process as the occluded object is hidden by overlaper.

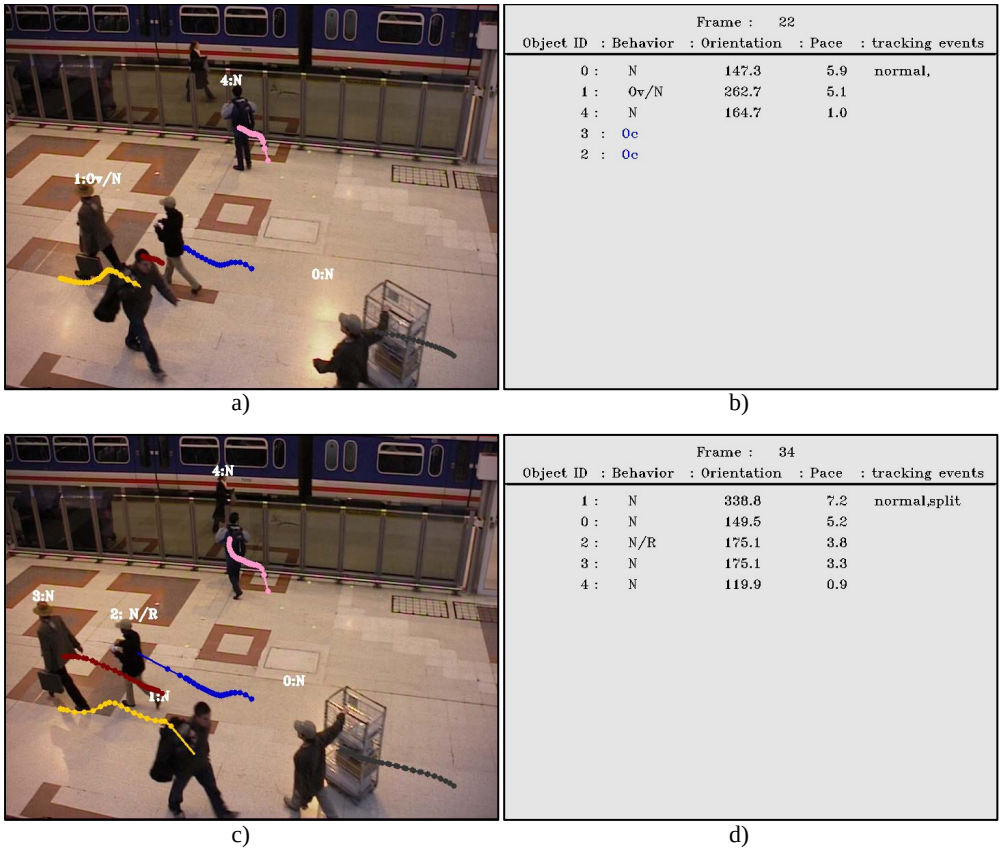


Figure 5.19: shows the results of tracking and behavior understanding on PETS2006 [1] dataset for *Frame 22* and *34*. a) and c) show the visual captured for *Frame 22* and *Frame 34* representing the occlusion and split situations of objects with ids *1*, *2* and *3*. In b) and d), *TnBII* panel demonstrates the state of the objects along with orientation and pace. Besides, it can also be noticed that both occlusion and split events are active in corresponding frames.

Frame 34: This scene indicates the split event which is observed when the occluded object reappears from the occlusion phase. The tracker re-estimates its path based on its own visual characteristics and physical location. The *TnBII* panel demonstrates the respective behaviors of objects during normal, occluded, and reappear situations.

Figure 5.20(a) and (b) demonstrates the occlusion and split situations as:

Frame 44: This scene shows the occlusion situation when two objects with ids *0* and *1* interact with each other. The object with id *0* is assigned overlaper behavioral state whereas the object with id *1* is assigned the occluded behavioral state. The tracker of occluded object is unable to advance localization.

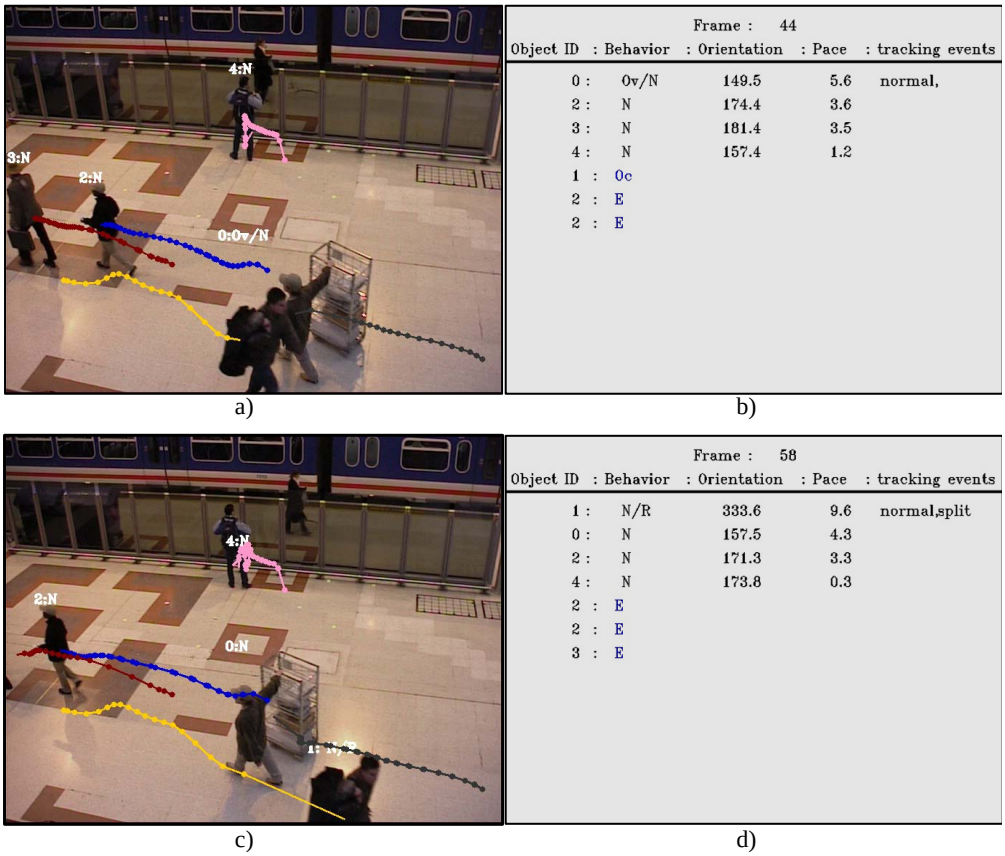


Figure 5.20: shows the results of tracking and behavior understanding on PETS2006 [1] dataset for *Frame 44* and *58*. a) and c) show the visual scene captured for *Frame 44* and *Frame 58* which are representing the occlusion and split situations of objects with id *0* and *1*. In b) and d), the *TnBII* panel demonstrates the state of the objects along with orientation and pace. It can also noticed that both occlusion and split events are active in corresponding frames.

Frame 58: This frame indicates that the split event is observed and the occluded object is reappeared from the occlusion phase. The tracker re-estimates its path based on its own visual characteristics and physical location. The *TnBII* panel demonstrates the respective behavior of objects during normal, occluded, and reappear situations. It is also noticeable that object with id *4* keeps its motion and state quite suspiciously in the scene.

Discussion

Due to high contrast and reflecting surface of the ground, strong shadows are appeared and detected as object itself. Moreover, appearances of objects are quite

similar (i.e., objects are wearing dark coats) which intensify load on our matching and identity management algorithm. However, despite of these issues, we are able to successfully perform the tracking and behavior understanding in a unified manner. In Appendix, Figure A.8 shows visual results of segmentation and feature detection.

5.6.3 PETS2009 Dataset

The third dataset used for testing is PETS2009 [3] dataset which is especially aimed for crowded scenes. This dataset is accessible publicly for tracking and behavior understanding tasks along with the information about the actions, locations, and behaviors of the actors. We have selected the dataset of the city center depicting various object behaviors, such as the random walk, standing in a group, and interaction with other passing objects (i.e., hand shake or meeting gestures). The scene is captured in many views but to achieve the tracking and behavior understanding, we have selected the view in which the visibility of objects are not very obscured. The PETS2009 [3] video sequences have been chosen for the following demonstration because it has many potential difficulties for people tracking:

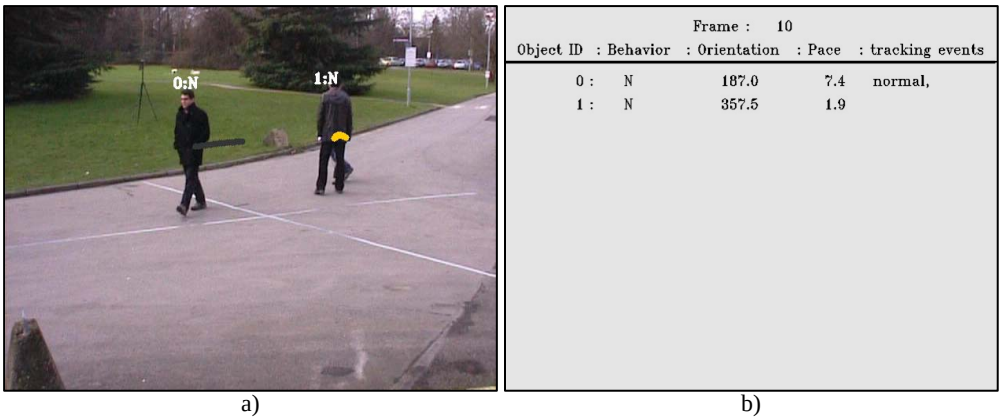
- stationary people standing on the way of the scene in the group,
- people vary the scale while walking,
- people occludes multiple times in the scene.

In Figure 5.21 shows the normal situation:

Frame 10: In this frame, there are three objects but one object is standing alone and assigned id *0*, and two people are standing in the group and assigned id *1*. Object with id *0* is walking in the forward direction where the object with id *1* is leaning in the scene, and currently in standing position as indicated by their pace and orientation in *TnBII* panel.

Figure 5.22(a) and (b) show the multiple occlusion and split situations as:

Frame 33: In this frame, objects with id *2* and id *0* occlude each other during the passage on their ways. The behavior of these objects are updated due to occlusion. Object with id *0* becomes overlaper and object with id *2* becomes occluded. As the contextual information is lost, the tracker of object with



a) b)

Figure 5.21: shows the results of tracking and behavior understanding on PETS2009 [3] dataset for *Frame 10*. a) shows the visual scene captured for *Frame 10* representing the ideal situation where three objects are detected and tracked. b) shows the *TnBII* panel demonstrating the information about behavior, orientation and pace of the object.

id *2* is unable to estimate the corresponding locations. This unusual situation (i.e., non-linearity) is handled by the behavioral states of the objects. The *TnBII* panel shows the corresponding information about the object behaviors and the tracking events.

Frame 41: This frame shows the split event where the occluded object with id *2* is reappeared again and retains its visual information. The tracker estimates the location based on the states prior to occlusion. The resulting track is treated by b-spline for smooth representation. The *TnBII* panel shows the corresponding information about object behaviors and the tracking events.

Figure 5.23(a) and (b) show the occlusion due to interaction and split situations as:

Frame 57: In this frame, objects with ids *1* and *3* are occluded due to usual hand shake and interaction. Object with id *1* becomes overlaper and object with id *3* becomes occluded. As the contextual information is lost, tracker is unable to estimate the corresponding locations of object with id *1*. The *TnBII* panel shows corresponding information about object behaviors and the tracking events.

Frame 71: In this frame, the objects are split and the occluded object with id *3* is reappeared again and retains its visual information. The tracker estimates the

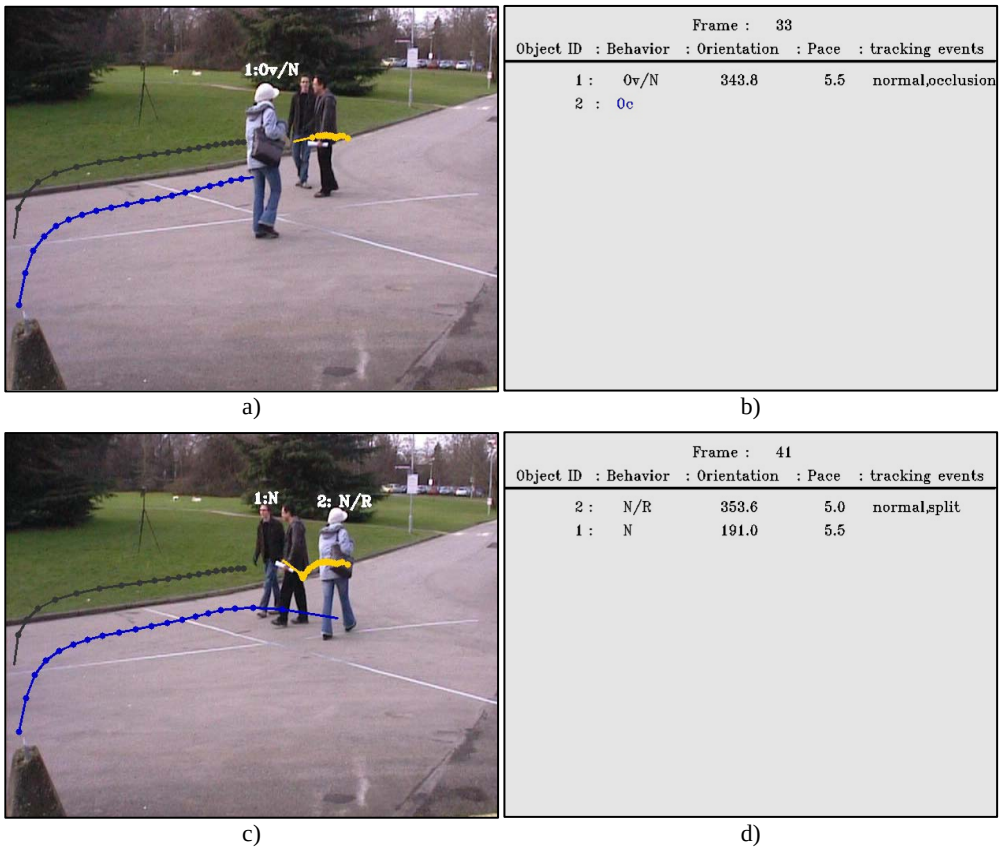


Figure 5.22: shows the results of tracking and behavior understanding on PETS2009 [3] dataset for *Frame 33* and *Frame 41*. a) and c) show the visual scene captured for *Frame 33* and *Frame 41* which are representing the occlusion and split situations of objects with ids *1* and *2*. In b) and d), the *TnBII* panel demonstrates the state of objects along with orientation and pace. It can also be noticed that both occlusion and split events are active in corresponding frames.

location based on the states prior to occlusion. The *TnBII* panel shows the corresponding information about object behaviors and the tracking events.

Discussion

The above sequence tries to depict the object behaviors in the places like city centers where the sequence containing objects with different behaviors. For instance, objects are standing in the group or interacting through hand-shake while crossing each other. This situation increases the level of complexity at various levels, such as object detection may fail when objects are standing for a longer period of times. The results demonstrate that both the tracking and behavior understanding during

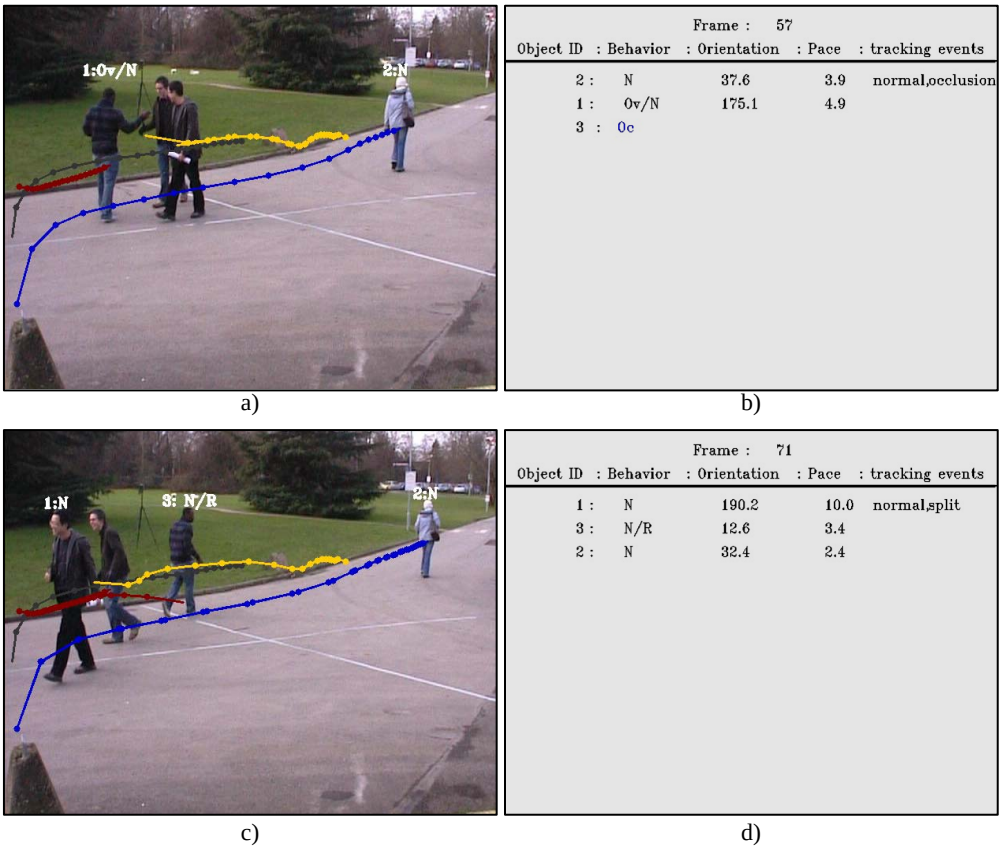


Figure 5.23: shows the results of tracking and behavior understanding on PETS2009 [3] dataset for *Frame 57* and *Frame 71*. a) and c) shows the visual captured for *Frame 57* and *Frame 71* which are representing the occlusion and split situations between objects with ids *1* and *3*. In b) and d), the *TnBII* panel demonstrates the state of objects along with orientation and pace. Besides, both occlusion and split events are active in corresponding frames.

motion is performed successfully along with other related information which is shown in *TnBII* panel at each time instance. In Appendix, Figure A.9 shows visual results of segmentation and feature detection.

5.6.4 Evaluation

We have evaluated our tracking and behavior understanding framework on the basis generating the correct identities and behaviors corresponding to objects during tracking. For such evaluation, the first essential requirement is ground truth. For this purpose, we have manually assigned the identities to objects and interpret their respective behaviors. Finally, the performance is evaluated by computing preci-

sion and recall measures. In the context of identities (i.e., analogous to tracks) and behaviors, precision and recall measures are defined as follows:

$$precision\ (pre.) = \frac{Number\ of\ correct\ identities\ or\ behaviors}{Number\ of\ established\ identities\ or\ behaviors}, \tag{5.25}$$

$$recall\ (rec.) = \frac{Number\ of\ correct\ identities\ or\ behaviors}{Number\ of\ actual\ identities\ or\ behaviors}, \tag{5.26}$$

where *actual identities or behavior* denotes the identities or behaviors available in the ground truth. Moreover, the evaluation is performed based on their ability to detect tracking events: 1) deal with entry and exit of objects, 2) handles occlusion event, and 3) handles the split event when objects are reappeared from occlusion.

Table 5.3: Precision and Recall of Tracking and Behavior Understanding Framework for IESK, PETS2006 [1] and PETS2009 [3] datasets

Dataset	**identities**		**normal**		**overlaper**		**occluded**		**reappear**		**new**		**exit**	
	pre.	rec.	pre.	rec.	pre.	rec.	pre.	rec.	pre.	rec.	pre.	rec.	pre.	rec.
IESK	0.88	0.89	0.94	0.90	0.91	0.91	0.81	0.85	0.85	0.9	0.83	0.79	0.65	0.83
PETS 2006	0.86	0.94	0.98	0.85	0.90	0.95	0.85	0.99	0.73	0.99	1	1	0.69	0.9
PETS 2009	0.90	0.89	0.80	0.81	0.88	0.79	0.91	0.90	0.90	0.90	1	1	0.91	0.91
Avg Result	0.87	0.89	0.91	0.85	0.90	0.88	0.85	0.91	0.83	0.93	0.94	0.93	0.75	0.88

Table 5.3 presents the performance values for the framework which are proposed for tracking and behavior understanding on test sequences. It is important to observe that the precision and recall of object identity recognition and normal behaviors are more prominent in performance. The precision and recall of the exit and new event are interrelated, because, if the object is wrongly classified as exit behavior then in the next instance, the algorithm will treat that object as a new. It is also observed that the performance of overlaper behavior is dominant over occluded behavior and it is due to the situation when the object features are overshadowed during the early phase of occlusion. The performance of reappear event is also linked to some other factors, such as if the object is misclassified during occlusion, then it will affect the reappear behavior as well. So, the recall is better

but precision is degraded.

Table 5.4 presents the performance values for the tracking event detection algorithm which is developed along with the tracking and behavior understanding framework. Infact, the detection of accurate events will lead to significant improvements in the results of Table 5.3, in some way. For instance, the axiom is called when a particular tracking event is activated. However, the mechanism of assigning identities, their management, and behavior inferencing is achieved by incorporating tracking axioms which control this type of discrepancy up to some extent in integrated qualitative and quantitative approaches. Moreover, it is observed that the recall values are dominant over precision because of many factors. For instance, mis-detections of objects may result in satisfying the condition for the exit events which consequently activates the new events in the next frames. However, the developed constraints for object states prevent these situations, but at the same time it affects the precision of the normal events (this event is defined when no conflicted events is observed).

Table 5.4: Precision and Recall of Tracking event detection for IESK, PETS2006 [1] and PETS2009 [3] datasets.

Dataset	**normal**		**occlusion**		**split**		**new**		**exit**	
	pre.	rec.	pre.	rec.	pre.	rec.	pre.	rec.	pre.	rec.
IESK	0.81	0.93	0.85	0.91	0.85	0.79	0.81	0.78	0.81	0.76
PETS2006 [1]	0.75	0.91	0.88	0.86	0.81	0.81	0.82	0.791	0.76	0.77
PETS2009 [3]	0.83	0.82	0.79	0.701	0.76	0.89	0.89	0.801	0.77	0.83
Avg Result	0.79	0.88	0.84	0.82	0.80	0.83	0.83	0.79	0.78	0.79

Remaining Problems

The following situations still pose a problem for proposed framework:

- Identifying Objects in Groups: The strategy of identity management of an object is, infact takes the input visual features driven from segmentation. As a result, it is impossible to assign the identities to every object walking in the group. For instance, two people walking side by side close to each other in a manner that the segmentation results in a form of the single blob. The

identity management scheme assigns all the objects in the group a single unique identity which will not be manageable if the objects are separated at a later time instance.

- Identifying Objects in Crowds: The strategy of the identity management and tracker are applied on normal and complex situations where the objects visual information is suffered from occlusion due to their intercepting pass ways in the image. In the presence of the crowd or in large groups of people who overlap in the image, it is impossible to continue individual object tracking.

- Regain the Objects Identities and Tracks: There are two types of limitations: first, if the object lost its identity, it is difficult to regain it at a later stage. Therefore, one solution is to register the object and then perform the simple comparisons for the identification of the object once again. Second, if the object has left the scene, and then it re-enters in the scene again, a new identity and tracker are assigned. The object will not be resuming with its old identity and tracker because we manage an identity pool and when an identity is freed, it is assigned to other objects. This is the reason that the reader will not find "fancy" numbers as identities to our objects.

5.6.5 Context of Use and Applicability

The proposed framework as the name suggests is aimed to not only track the objects in an unconstrained environment but also to interpret their respective behaviors and object specific information, such as the orientation of object or pace of the walk. While researching on this novel idea, we keep the certain context of application domain and are not limited to object surveillance only. For instance, the scene analysis can be performed to see when the objects are entering and how their behaviors are changing throughout their passage in the scene. Similarly, the system can provide assistance to the concerned users for tracing where the object is and if the occlusion is observed, then object should reappear after split. Besides, they can keep an eye on objects suspicious activities (e.g., strolling) to avoid the unlikely situations. Other domains of application can be explored by incorporating the contextual information and human assistance to construct the proposed framework for more practical real scenarios.

5.7 Discussion and Conclusion

In this chapter, we have presented the proposed framework for object tracking and behavior understanding in non-crowded scenes along with experimental results and performance evaluation. The proposed framework is comprised in a top to down modular hierarchy where the low level processing, such as object detection, and visual features is the first part. In parallel, another approach is introduced to detect tracking events which triggers corresponding axioms. At the core of this framework, we have described the integrated quantitative and qualitative approaches which combine the statistical measurements (i.e., BMW approach) with the logical models (i.e., tracking axioms). Each detected object contains a unique identity and behavioral states are maintained by corresponding approaches. Finally, Kalman filter based tracking system is developed that estimates the objects spatial locations over time. The experiments are conducted on three dataset of different nature to test the performance. Moreover, the ground truth is made to perform evaluation, and remaining issues are elaborated. In the end, we have described the context of use and degree of applicability of the proposed approach in real situations such as surveillance and behavior monitoring.

CHAPTER 6

Behavior Understanding in Crowded Scenes

In this chapter, we aim to investigate crowded scenes and propose a framework to understand the crowd behaviors. We begin with the definitions and terms used scientifically by the vision research community for crowd behavior analysis in Section 6.1. The proposed framework is presented in Section 6.2 which describes each of its components briefly. The core of the proposed approach is presented in Sections 6.3-6.7 to analyze the overall dynamics and to characterize the behaviors in distinct regions. In Section 6.8, we demonstrate the applicability of our proposed approach on the task of crowd behavior understanding. We have conducted experiments on two challenging benchmark crowd datasets PETS2009 [3] and UMN [2] to demonstrate the performances of the proposed framework. In addition, the comparative analysis, remaining problems, and context of applicability in real situations are also presented. This chapter closes with conclusive discussion in Section 6.9.

6.1 Crowds and Crowd Behavior Analysis

A crowd is defined as a place that contains a high density of objects. For vision community, the term "crowded scene" makes a generic reference to real-scene crowds. The constituted crowded scenes may contain a variety of objects and are not limited to people only, such as cars, flock of birds or a school of fish. Furthermore, the crowd itself can be categorized according to its corresponding dynamics into structured/coherent and unstructured/incoherent.

The crowd behavior understanding is a very broad term which in general refers to the inference of collective or individual behaviors of subjects forming crowd. However, the scope and context of crowd behavior understanding can be defined flexibly. Usually, there are many levels of crowd behavior understanding and analysis: 1) global level, 2) intermediate level, and 3) individual level as shown in Figure 6.1. One natural view of behavior understanding and anomaly detection is that we attempt to analyze the components which naturally characterize the "normal behaviors". In contrast, "anomaly or abnormal behaviors" detection refers to

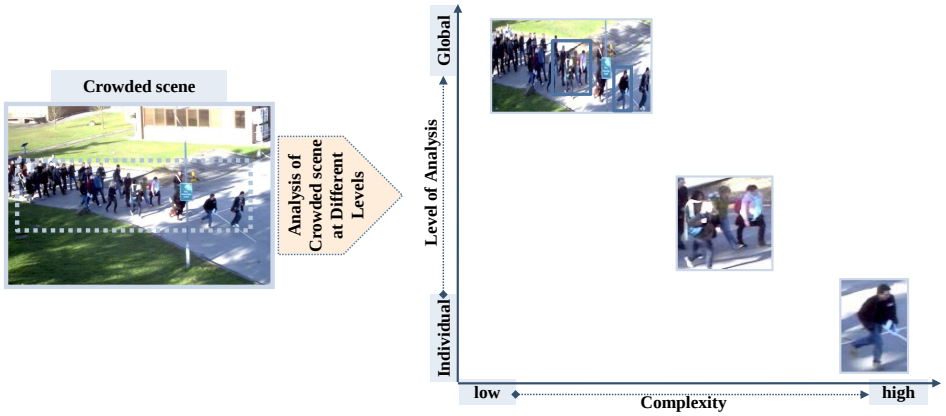


Figure 6.1: presents crowd behavior analysis at different level in graphical mode. Vertical axis defines the level of analysis whereas horizontal axis signifies the level of complexity. It is notable that the complexity increases from low to high as we move from global to individual level of behavior analysis.

the problem of pinpointing the locations that do not conform to normal behaviors or fall in its respective labeled class (i.e., abnormal). In literature, many techniques have been developed for anomaly detection based on certain threshold criteria and offer context specific solutions, while others suggested more generic ways by specifying each behavior with certain labels for example normal, abnormal, or specific categories of abnormality including running, dispersion, etc. Crowd behavior analysis and anomaly detection finds an extensive use in a wide variety of applications, such as localizing suspicious movement of individuals and assistance in emergency situations.

6.2 The Framework

Both abrupt activities and complex dynamics of individuals in crowd define collectively the self-organizing mechanism. To achieve the goals of crowd behavior understanding and anomaly detection, the proposed algorithm assume that the transport of individuals from one region to another is governed by optical flow in the video sequence. The idea is motivated by observing the motion perception phenomenon under the influence of optical flow that reveals the regions of qualitatively dominant dynamics. So, we can say that these regions have a direct correspondence with the distinct dynamics of crowds which emerge from the interactions of individuals with each other and with the environment.

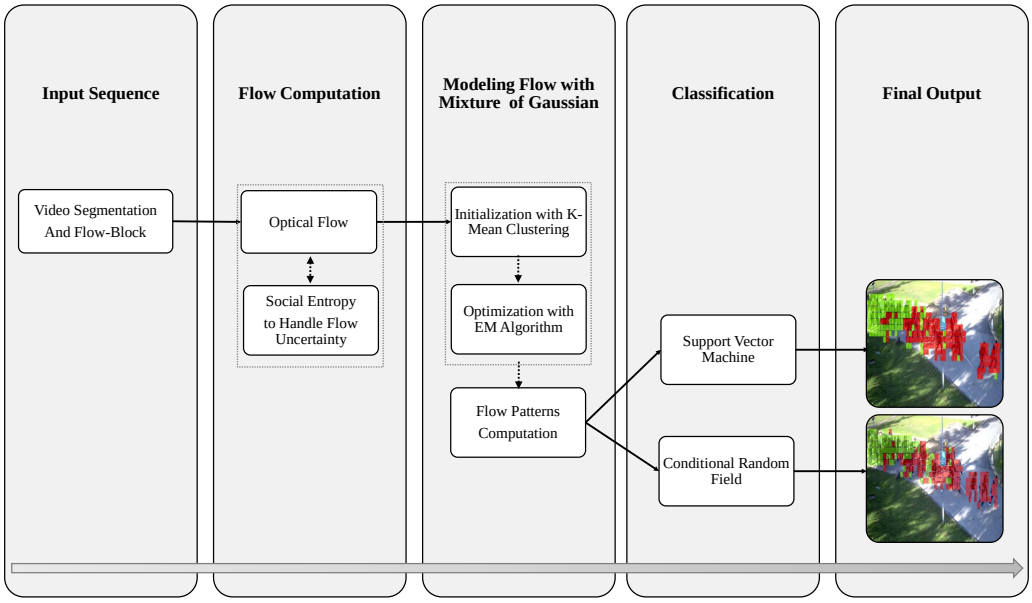


Figure 6.2: shows the proposed framework for crowd behavior understanding.

We propose a top to down approach which is staged in several phases to model and analyze the characteristics of crowd behaviors as shown in Figure 6.2. The proposed framework has four main modules: i) video segmentation and flow-block formation, ii) flow computation and uncertainty handling with social entropy, iii) modeling the flow with mixture of Gaussians, and iv) behavior classification. In the following, we have briefly described these modules and later explained them in the subsequent sections.

Video Segmentation and Flow-Block Formation: Our proposed approach begins by extracting the foreground through segmentation. The video sequence is windowed into overlapping but fixed size segments which we referred as video segments. Each video segment is spatially divided into non-overlapping blocks which we referred as flow-blocks[1].

Flow Computation and Uncertainty Handling with Social entropy: The optical flow is computed at frame level (i.e., globally) over the foreground. Next, we make use of recent advances in the areas of social entropy for handling optical flow uncertainties. The concept of social entropy is originated from the field of social sciences and used in Social Entropy Theory [14]. Social en-

[1]Both block and flow-block terms are used interchangeably, unless mentioned.

tropy empirically determines a quantitative metric that enables us to extract the refined optical flow.

Modeling Flow with Mixture of Gaussians: The idea is to use mixture of Gaussians to uncover the spatial organization of the flow cloud data in each flow-block. At the conceptual level, the implication of using mixture of Gaussians is to parameterize the computed flow cloud data in flow-block which helps in assimilating the motion information in spatio-temporal space. In practical terms, mixture of Gaussians quantifies and prototypes the flow cloud data observed in flow-block over an interval (i.e., temporal space of flow-block) containing significantly correlated and uncorrelated flow cloud data. Therefore, it helps in revealing the representative characteristics of the underlying dynamics, such as static objects moving with different paces. As, the flow field is directly related to the dynamics of individuals in crowd, these characteristics have a direct relationship with the behaviors of objects in crowded scenes. So, these mixture of Gaussians distributions result in the distinct flow patterns referred to as feature vectors for flow-block.

Behavior Classification: We have treated the crowd behavior understanding and anomaly detection as two class problem. But a fundamental question is how to model feature vector representing flow patterns in each flow-block. For this purpose, we have performed classification first by using Support Vector Machine (SVM) [11]. In the next phase of experiments, we have used Conditional Random Field (CRF) to localize the crowd behaviors. The performance of both classification schemes are compared both quantitatively and qualitatively.

6.3 Video Segmentation and Block Creation

Given a video sequence E with K frames, the first task is to perform segmentation and extract foreground of the scene. We have used the suggested approach for segmentation as described in Section 3.1 and an example of extracted foreground is shown in Figure 6.3. As, it is observed in crowded scenes, the occupancy region at every frame is important and provides the distinctive attributes. So, we begin by marginalizing the video sequence into equally sized segments (i.e., video segments) as presented in Figure 6.4(a). The selection of segment size depends upon the

Figure 6.3: shows an example of detected foreground in the PETS2009 [3] crowded scenes.

dataset and frame rate of the video sequence. In our case, we kept the size (i.e., $seg_{size} = 3$) for each video segment S_n and the segmented video V is as follows:

$$V = [S_n; n = 1, \ldots, K - seg_{size} + 1]; \tag{6.1}$$

Our next objective is to create flow-blocks for each video segment (S_n). For this,

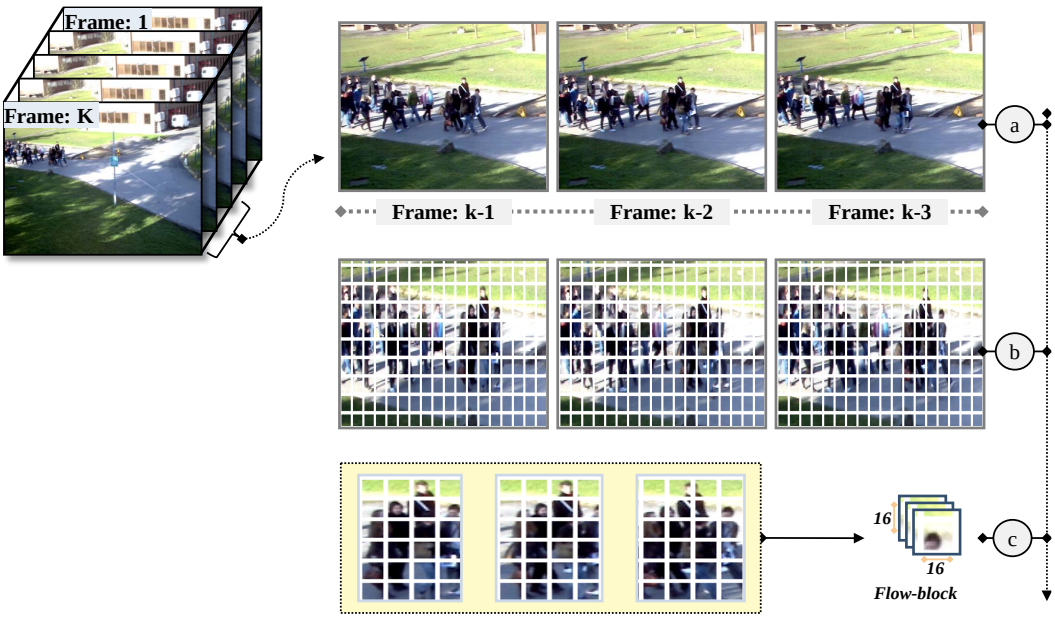


Figure 6.4: presents the different level of algorithm process performed on PETS2009 [3] dataset. For instance, a) shows the video segment containing three consecutive frames in it, b) presents the process of fixed size block formation over the frame, c) illustrates the flow-block creation.

we have divided every video segment into M fixed size non-overlapping patches[2],

[2]The size of non-overlapping patch is selected (i.e., $16 \times 16 \times seg_{size}$) after conducting empirical studies over the dataset (i.e., PETS2009 [3]).

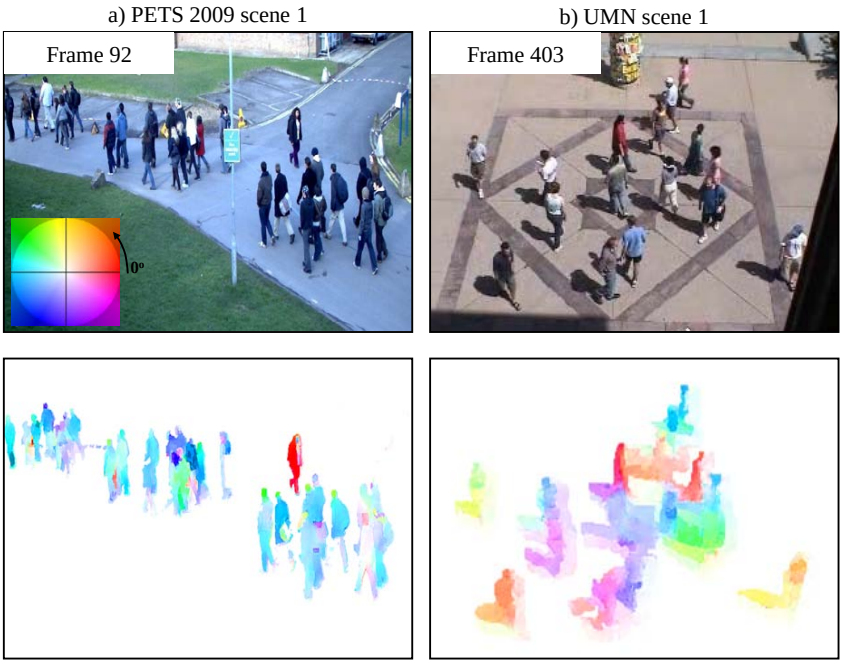


Figure 6.5: shows the results of optical flow approach [5]. The sequences are taken from PETS2009 and UMN which indicate both normal and abnormal situations. The flow field is mapped using the color wheel encoding scheme [102] to indicate its strength.

named as flow-blocks. These flow-blocks are obtained inside each video segment in Figure 6.4(b-c) and described as follows:

$$S_n = \left\{ F_{(m,l)}; m = 1, \ldots, M; l = 1, \ldots, seg_{size} \right\}; \tag{6.2}$$

where each video segment S_n contains M flow-blocks $F_{(m,l)}$.

6.4 Optical Flow Computation and Social Entropy

We have computed optical flow as described in Section 4.2 between the consecutive frames of the given video sequence. The optical flow technique considers both the global and local aspects of grey value constancy, gradient constancy, smoothness, and multi-scale constraints to estimate the optical flow. Figure 6.5 shows color-coded optical flow computed from different sequences in our datasets.

Given a crowded sequence of K frames, the optical flow is computed at frame level over the extracted foreground and a flow cloud data is generated for each

flow-block. So, each flow-block contains the computed flow at P locations (i.e., $16 \times 16 \times seg_{size}$) which is defined as follows:

$$F_{(m,l)} = (f_p; p = 1, \ldots, P), \tag{6.3}$$

$$\vec{f}_p = (v_x, v_y), \tag{6.4}$$

where flow-block $F_{(m,l)}$ contains P flow cloud data computed at frame level and treated at flow-block level for further modeling, each flow field contains flow velocities $\vec{f}_p = (v_x, v_y)$. As indicated, v_x and v_y represent the velocities along with the horizontal and vertical axis of the motion field.

Handling Optical Flow Uncertainties with Social Entropy

Analyzing the pixel movements in a video sequence allows the inference of overall motion that includes the motion of object and self-motion of the image capturing device. In general, these motions are estimated by processing spatial and temporal derivatives of image values, for instance pixel intensities which hold the motion information of image structure. However, the measured motion of given images is noisy due to the transformation of 3D real scene onto 2D image which is based on some approximations, for instance, the physical relationship between spatio-temporal image values and motion due to self-movement within 3D environment. The approaches suggested for optical flow analysis should be able to cope with: 1) correspondence problems due to ambiguities (e.g., periodicity or lack of texture) in the image structure, 2) camera noise, 3) spatial motion discontinuities, and 4) temporal movement changes.

We have employed Anisotropic Huber-L1 approach [5] which is based on combining data by assuming fundamental constancy of image property (e.g., brightness) [43] spatially. It is a spatio-temporal regularization approach in which the expected flow across the image is modeled by replacing the isotropic TV regularization with an anisotropic Huber regularization. However, the choice of parameters is crucial because it is difficult to trace the influence [111] on the accuracy of optical flow for a wide range of video sequences. Therefore, we have addressed the optical flow uncertainty by incorporating an explicit approach called social entropy. Moreover, we have given a generic formulation and its concept to deplete the flow field uncertainties observed when incorporating social entropy which faithfully reveals the characteristics of crowd dynamics.

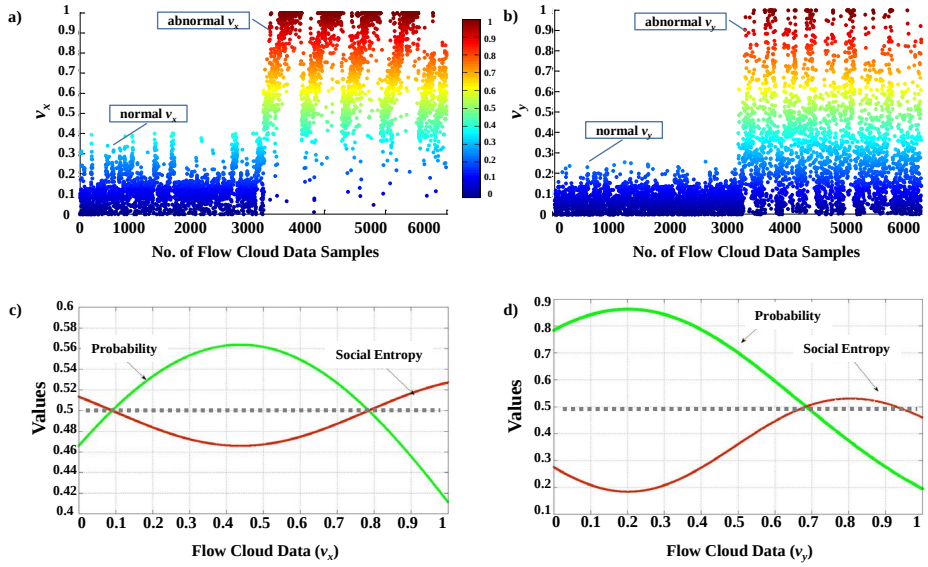


Figure 6.6: shows the computation of social entropy over training dataset of PETS2009 [3] dataset for normal and abnormal behaviors. a) and b) show the values of flow field of training samples. c) and d) present the probability (i.e., green curve) P_i and the corresponding Shannon entropy (i.e., red curve) H_i of the flow field $\vec{f}_p$.

Social entropy[14] empirically determines a quantitative criteria and is used as an optimization strategy to handle the uncertain flow field distribution. Given the training datasets (i.e., for normal behavior and abnormal behavior) as shown in Figure 6.6 (a-b) for crowd behavior understanding, we have first measured the probability of flow cloud data containing velocities in horizontal and vertical directions in Figure 6.6(c-d). These distributions uncover the homogeneity and heterogeneity in the flow cloud data of the training dataset in Table 6.1. After that, social entropy is measured in Figure 6.6(c-d) over this distribution of flow cloud data to reveal their uncertain characteristics which allows us to define the criteria for uncertain data in flow fields. Mathematically, the social entropy of the flow cloud data is described as Shannon [112] information entropy H_i:

$$H_i = -P_i log_2 P_i; \tag{6.5}$$

where i is the number of flow samples, P_i is the probability of the flow field and H_i is the corresponding entropy measure.

It is observed that both probability distribution and entropy of corresponding flow cloud data is monotonically inversely proportional to each other. The low en-

tropy H_i is observed in the distributions P_i which are sharply peaked around few values, whereas the values which are spread more evenly have higher entropy values H_i. Based on these statistical analysis, we have obtained a criteria as marked by grey line in Figure 6.6(c-d) which substantiates the reliable and unreliable flow cloud data which can either be removed or shifted to low entropy range. We argue that the measured entropy of flow cloud data reflects the uncertain optical flow computed from the suggested method which results in the characterization of incorrect crowd behaviors.

Incorporating Social Entropy During Testing. In the above, we have learned and measured the social entropy for flow cloud data which represents both normal and abnormal behaviors of crowded scenes as shown in Figure 6.6. During the testing, prior to flow modeling with mixture of Gaussians, we first handle the flow uncertainties. Each of our flow-block (i.e., $F_{(m,l)}$) is treated as an independent social system, so, the 2D distribution of $\vec{f}_p$ in each flow-block either represents certain or uncertain flow cloud data marked as grey line in Figure 6.6. As, observed in the figure, we consider the flow field value below the grey line as certain and keeping them for flow modeling whereas the uncertain flow field is treated as noise (i.e., uncertain) and thus is not considered for flow modeling.

6.5 Modeling Flow with Mixture of Gaussians

In common vision practice, mixture of Gaussians acts as mode finding approach to model feature vectors associated with each pixel (e.g., position, color, flow) which are taken as samples from an unknown probability density function and the clusters are formed in this distribution. Based on this, we have used this property of mixture of Gaussians to obtain a set of feature vectors instead of modeling the feature space (i.e., flow cloud data) for analyzing the crowd behaviors directly. The main concept lies on the fact that the given flow cloud data in each flow-block are significantly different and correlated which are required to be gleaned prior to model as feature vectors by applying parametric approximation.

Let us consider a cloud of flow field in flow-block as shown in Figure 6.7. How would you learn and classify the behaviors of crowd based on these flow fields in each flow-block alone? There can be many possible ways, for instance taking the mean of the flow field which returns a value but not a feature, so the

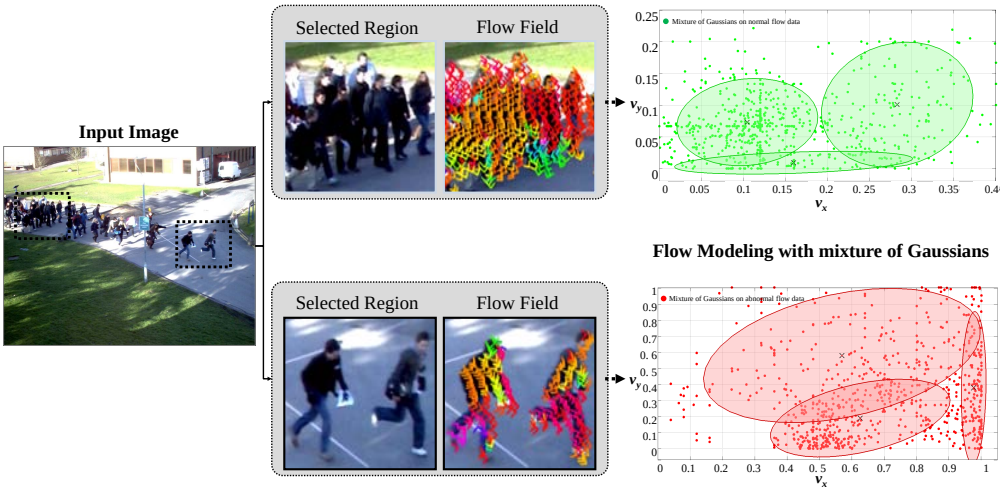


Figure 6.7: shows the modeling flow cloud data computed on sample frame of PETS2009 [3] dataset with mixture of Gaussians. In this, we have selected a region and optical flow is computed over it. Next, the flow field is plotted and the mixture of Gaussians is fitted. The green color shows the normal flow cloud data and red color represents the abnormal flow cloud data.

compromise is on the precision. In Figure 6.7, we have presented a visualization of flow field $\vec{f}_p$. How many obvious flow patterns do you see? How would you interpret the underlying behaviors from these flow cloud data alone? Therefore, without losing generality of flow field in each flow-block, we have learned and fitted mixture of Gaussians as flow field representatives. Moreover, the feature vectors (i.e., flow patterns) are computed from these mixtures and are characterized by the classification approaches.

We define the cloud of flow field (i.e., $\vec{f}_p$) in each flow-block (see Equation 6.3) as 2D random samples which are extended over the spatio-temporal range as shown in Figure 6.7. So, given our 2D distribution of flow field in each flow-block, K-means clustering algorithm is employed to initialize the model and to find the optimal number of mixtures. Next, EM approach is used as an optimization function for finding the maximum likelihood solutions for our distributions, iteratively. So, we have used this clustering approach to iteratively re-estimate the parameters of corresponding mixture of Gaussians for density function and is written as:

$$P(\vec{f}_p) = \sum_{c=1}^{C} w_c \mathcal{N}(\vec{f}_p | \vec{\mu}_c, \Sigma_c); \tag{6.6}$$

where c represents the mixture of Gaussians, w_c is the mixing coefficient, $\vec{\mu}_c$ is the

mean, covariance Σ_c matrix and are the parameters of each component of Gaussian model in respective order.

The EM algorithm works in two stages to iteratively compute the maximum likely estimate for the unknown mixture parameters $\{w_c; \vec{\mu}_c; \Sigma_c\}$:

E-Step: Expectation stage is used to estimate how likely a flow field sample $\vec{f}_p$ is generated from c-th Gaussian and estimates the responsibilities:

$$z_{pc} = \frac{1}{Z_p w_c} \mathcal{N}(\vec{f}_p | \vec{\mu}_c, \Sigma_c), \qquad with \ \sum_c z_{pc} = 1, \tag{6.7}$$

where z_{pc} is the cluster label for data points and Z_p is the missing or unobservable samples.

M-Step: Maximization stage is used to update the parameter values and to estimate the number of samples $N_c = \sum_p z_{pc}$ assigned to each cluster.

$$\vec{\mu}_c = \frac{1}{N_c} \sum_p z_{pc} \vec{f}_p, \tag{6.8}$$

$$\Sigma_c = \frac{1}{N_c} \sum_p z_{pc} (\vec{f}_p - \vec{\mu}_c)(\vec{f}_p - \vec{\mu}_c)^T, \tag{6.9}$$

$$w_c = \frac{N_c}{N}, \tag{6.10}$$

The computed parameters of c mixture of Gaussians for each flow-block presents a concrete representation of flow cloud data characteristics as presented in Equation 6.6 whereas Figure 6.7 illustrates graphically the mechanism. The parameter of mixtures, particularly $\vec{\mu}_c$ presents the mean in each dimension of flow field cloud bounded by corresponding Gaussian components (i.e., $\mu_{c_{vx}}$ and $\mu_{c_{vy}}$). So, we have computed the mean density (i.e., $d_{\vec{\mu}_c}$) for each Gaussian resulting in the sequence of flow patterns for each flow-block, which is then given to classifiers (i.e., SVM and CRF) for detecting the normal and abnormal behaviors. The size of the of flow patterns (i.e., FV or $\vec{x}$) is directly related to the number of mixtures C. We can write it as:

$$\vec{\mu}_c = (\mu_{c_{vx}}, \mu_{c_{vy}}), \ \ and \quad d_{\vec{\mu}_c} = \sqrt{\mu_{c_{vx}}^2 + \mu_{c_{vy}}^2}, \tag{6.11}$$

$$\vec{x} = \left\{ d_{\vec{\mu}_c}; c = 1, .., C \right\}, \tag{6.12}$$

6.6 Behavior Analysis with Support Vector Machine

We have employed structured SVM [11] for crowd behavior analysis which has many good practical and theoretical properties where it allows us to handle crowd datasets (i.e., PETS2009 [3] and UMN [2]). We define $\vec{x} \in X$ as an input sequence (i.e., $\vec{x} = x_1, \ldots, x_n$) of flow patterns in Equation 6.12 and $\vec{y} \in Y$ is the corresponding label sequence (i.e., $\vec{y} = y_1, \ldots, y_n$) of normal and abnormal behavior classes. So, the approach we pursue is to learn a discriminant function $F : X \times Y \longrightarrow \mathcal{R}$ over input-output pairs from which we can derive a prediction by maximizing F over the response variable for a given input $\vec{x}$. Hence, the general form of our hypotheses f is defined as:

$$f(\vec{x};\vec{w}) = \underbrace{argmax}_{\vec{y} \in Y} F(\vec{x},\vec{y};\vec{w}); \tag{6.13}$$

where $\vec{w} \in \mathcal{R}^n$ is the parameter vectors of model, F acts as linear function and measures the compatibility of input $\vec{x}$ and output $\vec{y}$ pairs:

$$F(\vec{x},\vec{y};\vec{w}) = \langle \vec{w}, \Psi(\vec{x},\vec{y}) \rangle ; \tag{6.14}$$

where $\Psi(\vec{x},\vec{y})$ is the combined feature representation of input $\vec{x}$ and output $\vec{y}$.

For training the weights $\vec{w}$ of the linear discriminant function, the standard SVM [11] optimization problem can be generalized in several ways where $n-slack$ formulations are commonly used which assign a different slack variable to each of the n training examples. We have used slack-rescaling in which the slope is adjusted while the position of the hinge is fixed [11].

We have employed Radial Basis Function (RBF) as a kernel because it does not require the feature space in its explicit form. It is due to the fact that only the inner products between support vectors and vectors of the feature space are sufficient. Therefore, the problem that arises from the higher dimensional feature space is alleviated because it allows the computations to take place in the original feature space. The use of kernel functions is usually referred as "kernel trick". In the generalized formulation of inner product in the joint representation, the joint kernel function [113] is written as:

$$J((\vec{x},\vec{y}),(\vec{x}',\vec{y}')) = \left\langle \Psi(\vec{x},\vec{y}), \Psi(\vec{x}',\vec{y}') \right\rangle ; \tag{6.15}$$

Once kernel SVM [11] is trained on a scan $(\vec{x},\vec{y})$, we can find the most probable

labeling of Equation 6.13 for a scan $\vec{x}^{'}$ where α is a vector of dual variable. This process is defined as:

$$\underbrace{argmax}_{\vec{y}^{'}} \sum_{\vec{y}^{'}} \alpha_{\vec{y}^{'}} [J(\vec{x}^{'},\vec{y}^{'},\vec{x},\vec{y}) - J(\vec{x}^{'},\vec{y}^{'},\vec{x},\vec{y})]; \tag{6.16}$$

In order to use combinatorial optimization , a RBF kernel should be decomposable to factors [114] and can be written as:

$$J(\vec{x},\vec{y},\vec{x}^{'},\vec{y}^{'}) = \sum_{j=1}^{J}\sum_{n=1}^{N}\sum_{n^{'}=1}^{N} exp(-\gamma||x_n - x_n^{'}||^2) y_n^j y_{n^{'}}^j + \tag{6.17}$$
$$\sum_{j=1}^{J}\sum_{l=1}^{J}\sum_{(n,m)}\sum_{(n^{'},m^{'})} exp(-\gamma||x_{nm} - x_{n^{'}m^{'}}^{'}||^2) y_n^j y_m^l y_{n^{'}}^j y_{m^{'}}^l;$$

where $(n,m) \in \xi$ and $(n^{'},m^{'}) \in \xi^{'}$ are the scan edges $\vec{x}$ and $\vec{x}^{'}$ and γ is set to 1.0. Using structured SVM [11] as a classifier with our flow patterns (i.e., $\vec{x}$) in Equation 6.12, we can distinguish specific and overall crowd behaviors. As, discussed earlier, the computed flow feature vectors reveals the reliable characteristics in the scene, which corresponds to the respective crowd behaviors.

6.7 Behavior Analysis with Conditional Random Field

Conditional Random Field is a probabilistic framework for inferencing a particular label sequence given the observation sequence, a detailed description is presented by Lafferty et al. [12] on CRF. Particularly, in our case, $\vec{x}$ is input sequence (i.e., $\vec{x} = x_1,\ldots,x_n$) of n flow patterns in Equation 6.12 and $\vec{y}$ is the corresponding label sequence (i.e., $\vec{y} = y_1,\ldots,y_n$) of normal and abnormal behavior classes. We assume that both sequences $\vec{x}$ and $\vec{y}$ are of same length. The probability of label sequence $P(\vec{y}|\vec{x};\theta)$ given the observation sequence [12] is defined as:

$$P(\vec{y}|\vec{x};\theta) = \frac{1}{Z(\vec{x},\theta)} \exp \sum_i \theta_i F_i(\vec{x},\vec{y}); \tag{6.18}$$

where the numerator $F_i(\vec{x},\vec{y})$ is the feature function which represents the paired mapping $F_i : X \times Y \longrightarrow \mathcal{R}$ of the data space X and the label space Y at the different levels of granularity. Therefore, feature function F_i can be arbitrarily corre-

lated [12] and is defined as follows:

$$F_i(\vec{x},\vec{y}) = \sum_j f_i(y_{j-1}, y_j, \vec{x}, j); \tag{6.19}$$

where f_i is the low level feature function which is influenced by the subset of the above entities such as, previous label y_{j-1}, current label y_j, observation sequence $\vec{x}$, and current position j.

The denominator $Z(\vec{x},\theta)$ in Equation 6.18 is the partition function commonly termed as normalization factor which ranges over all the label sequence but we assume here that the feature-function depends on at most two labels. So, instead of enumerating all possible $\vec{y}$, this assumption allows us to enumerate the possible $\vec{y}$ efficiently. The formulation of Z [12] is as follows:

$$Z(\vec{x},\theta) = \sum_{\vec{y}} \exp \sum_i \theta_i F_i(\vec{x},\vec{y}); \tag{6.20}$$

Training CRF. We perform the training using stochastic gradient methods based on the gradient of conditional likelihood function for nonlinear optimization. The goal of this learning task is to compute parameter θ (i.e., weights) values of our model and learns the conditional log-likelihood (CLL) of the training sequences and so our objective here is to maximize the CLL. For this purpose, among many sophisticated techniques, we used stochastic gradient ascent method for training [12]. The formulation is defined in the following:

$$\frac{\partial}{\partial \theta_i} \log p(\vec{y}|\vec{x};\theta) = F_i(\vec{x},\vec{y}) - \frac{\partial}{\partial \theta_i} \log Z(\vec{x},\theta); \tag{6.21}$$

In the above equation, for each θ_i, the partial derivative of CLL is evaluated for a single training sequences (i.e., one weight for each feature-function). Precisely, the partial derivative with respect to θ_i is the i-th value of the feature function for its true label $\vec{y}$, minus the averaged feature-function values for all possible labels $\vec{y}'$ (i.e., E). So, the above equation can be rewritten as:

$$\frac{\partial}{\partial \theta_i} \log p(\vec{y}|\vec{x};\theta) = F_i(\vec{x},\vec{y}) - E_{\vec{y}' \approx p(\vec{y}'|\vec{x};\theta)} \left[F_i(\vec{x},\vec{y}') \right]; \tag{6.22}$$

In practice, the function $\log(\theta)$ does not maximize in a closed form solution therefore, we invoke BFGS (Broyden Fletcher Goldfarb Shanno) as an optimization

routine that estimates the curvature numerically from the first derivative of the CLL and avoids the requirement of exact Hessian inverse computation with stochastic gradient ascent [12].

Inferencing CRF. Given the test sequence of flow patterns for each flow-block $\vec{x}$ and the learned parameter values of θ from the training data, the corresponding label ($\vec{y^*}$) for the sequence is obtained as:

$$\vec{y^*} = \underbrace{argmax}_{\vec{y}} p(\vec{y}|\vec{x};\theta) = \underbrace{argmax}_{\vec{y}} \sum_i \theta_i F_i(\vec{x},\vec{y}); \tag{6.23}$$

Using the definition of feature function [12] in Eq.6.19, we get:

$$\vec{y^*} = \underbrace{argmax}_{\vec{y}} \sum_i \theta_i \sum_j f_i(y_{j-1}, y_j, \vec{x}, j); \tag{6.24}$$

Each label sequence is aggrandize from $< start, end >$ states of labels (i.e., y_0 to y_{n+1}), so for the efficient computation, an alternative choice is to employ matrices. For this, g_j is a $q \times q$ matrix where q is the cardinality of the set vectors in the label sequence $\vec{y}$ and is defined over each pair of labels y_{j-1} and y_j [12] as follows:

$$g_j(y_{j-1}, y_j|\vec{x}) = \exp(\sum_i \theta_i f_i(y_{j-1}, y_j, \vec{x}, j)); \tag{6.25}$$

So, for each j, we will get different g_j functions which depends on weight θ, test observation sequence $\vec{x}$ and the position j. The sequence probability of the label $\vec{y}$ given observation sequence $\vec{x}$ can be rewritten in compact manner in the following:

$$P(\vec{y}|\vec{x};\theta) = \frac{1}{Z(\vec{x},\theta)} \prod_j g_j(y_{j-1}, y_j|\vec{x}); \tag{6.26}$$

$$Z(\vec{x},\theta) = \prod_j g_j(y_{j-1}, y_j); \tag{6.27}$$

Our main aim in obtaining the flow patterns for each flow-block is that, it is difficult to reveal the required level of details which can differentiate the coherent and incoherent dynamics at the global level. Therefore, the flow patterns obtained through modeling with mixture of Gaussians faithfully characterizes the behavior of the crowd dynamics which are modeled with CRF [12] to characterize the normal and abnormal behaviors in the crowd physics.

A bank of CRF [12] models is constructed, one for each flow-block to model the

flow patterns with corresponding label sequence and to characterize the crowd behavior at the specific and global level in an unconstrained environment. In contrast, generative modeling approaches [89] [59](i.e., HMM and LDA) require stringent conditional independence among the observed flow fields for more tractable joint distributions. Moreover, the evaluation is based on the two benchmark datasets by PETS2009 [3] and UMN [2] whereas the comparative analysis is performed with two related literatures [93] [59] addressing the similar problem.

6.8 Experiments and Discussion

This section presents experimental setup and datasets used in the experiments (see Appendix A.5 for more results). In addition, we have demonstrated the results of behavior analysis and anomaly detection along with a discussion and the context of applicability.

Table 6.1: Training process of PETS2009 [3] dataset

training scenario	training set (time stamps)	total training frames
Dataset S1, Level 1	13-57	220
Dataset S1, Level 1	13-59	240
Dataset S1, Level 2	14-06	200
Dataset S1, Level 3	14-17	90
Dataset S1, Level 3	14-33	343

6.8.1 Data Preparation, Train and Test Process

The proposed approach is tested on two publicly available benchmark datasets from PETS2009 [3] and UMN [2]. The first dataset used for the development of ideas is taken from PETS2009 [3] dataset. The PETS series of workshops make available public datasets for the comparison of tracking and surveillance technologies. These datasets are available with the information about the behaviors of the actors contained within and came up with new motivations and challenges to handle crowded scenes. The dataset comprises of three categories along with specific tasks for each category. For example, Dataset1 (i.e., S1) is used for person count and density estimation of crowds, Dataset2 is focused to perform tracking on objects in crowded scene, and Dataset3 (i.e., S3) is aimed to perform flow analysis of crowd and to

determine the respective events. This research is aimed to perform behavior analysis and anomaly detection using Dataset3 (i.e., S3) on PETS2009 [3] dataset test sequences in Table 6.2. PETS Dataset1 (i.e., S1) is used for training in Table. 6.1 which indicates the scenarios and the datasets used for the training process along with creating our own artificial dataset for training the abnormal situations in a similar way as suggested in [115]. The normal situations are represented by the usual walk of large number of people whereas the corresponding abnormal situations (i.e., running, panic and dispersion) are observed when individuals or a group of individuals deviate from the normal behavior. The dataset has been chosen because it has many potential difficulties:

Table 6.2: Testing sequences of PETS2009 [3] and UMN [2] dataset

testing scenario	testing set (time stamps)	total testing frames
Dataset S2, Level 1	13-57	400
Dataset S3, Level 3	14-16	100
Dataset S3, Level 3	14-31	380
Dataset S3, Level 3	14-33	100
Dataset UMN Seq 1	not given	400
Dataset UMN Seq 2	not given	390
Dataset UMN Seq 3	not given	300
Dataset UMN Seq 4	not given	500
Dataset UMN Seq 5	not given	400

- scene contains disturbances from external sources, for instance, camera flickering or weather conditions, and
- the objects behaviors are transformed from start till end. The starting frames contain few people, which gradually increase as the time passes. In the similar manner, objects in crowd do not adapt to sudden changes in behaviors but in a gradual manner.

The second dataset is from the UMN [2] dataset test sequences in Table 6.2 containing indoor and outdoor crowded scenes with normal and abnormal (i.e., run or dispersion) behaviors. The scenes were filmed in open garden, sitting and foyer places from the top view and contains about 25 to 30 actors moving in random direction and with random pace. Due to the long duration of these clips and low resolution of scene, the performance of our algorithm is fast. The dataset has been chosen because it has many potential difficulties:

- the scene is taken from top view where the observed flow field have very low values, and

- the characterization of crowd behavior is difficult when the objects are moving in parallel to camera view.

There is a major distinction between these two datasets, for example, in PETS2009 [3], the abnormality begins gradually unlike UMN [2] dataset, which makes PETS more challenging due to the transitions from normal to abnormal situations.

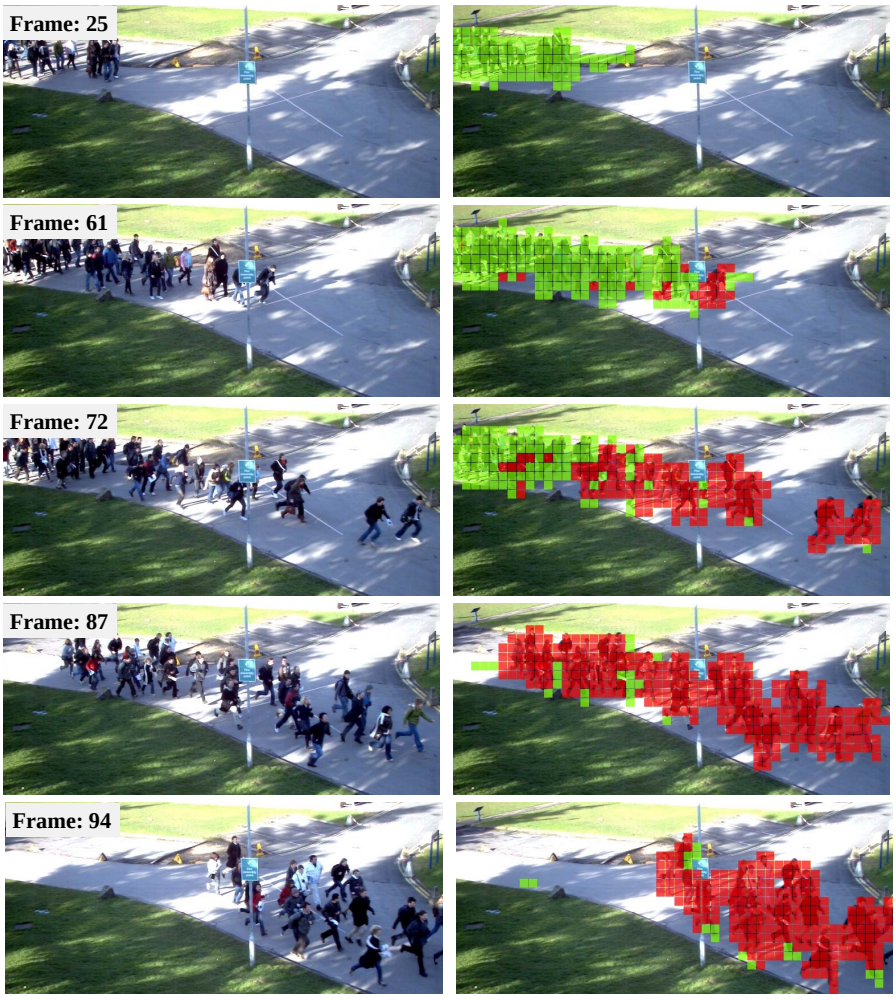


Figure 6.8: shows the detection results on PETS2009 [3] sequence. The normal behaviors are indicated by green blocks and abnormal behaviors are marked with red blocks.

6.8.2 Experiments with SVM

This section presents the qualitative analysis performed using structured SVM [11] classification over test sequences in Table 6.2. The behaviors are visualized for each flow-block where green patches indicate the normal and red patches show the abnormal behaviors (see Appendix Figures A.12-A.15). Moreover, the trends of events for complete sequence are also presented.

Qualitative Analysis

Figures 6.8 and 6.9 show the behavior understanding results on the selected frames of sequences from PETS2009 [3] datasets.

The first sequence, shown in Figure 6.8 depicts a group of people moving across the walking track of the scene. In this sequence, people are entering in the scene with common dynamics and desirable goal at *Frame 25*. But, after some instance of time, the leading persons in the group start running at *Frame 61*. As, the people are walking with some specific goal, so the people following the leading members also start running at *Frame 72* and *Frame 87*. As a consequence, the common dynamics of the group is now in transition (i.e., normal to abnormal) where some people (i.e., in the end) are still walking whereas the people at the front are running. Some instances later, we have seen that, at *Frame 94*, all the people are running in similar direction and once again the crowd is in common dynamics but with different behavior (i.e., abnormal or run).

In the sequence shown in Figure 6.9, people are entering in the scene from various directions with desirable goal to gather at the center of the scene in *Frame 25*. The group of people remains gathered for sometime in *Frame 51*. But, at *Frame 294*, it is shown that some people start running at random. As soon as, the other people observed this panic, they have also started running in random directions in *Frame 299*. The common dynamics of the gathered group is now turned into dispersion where the people are running in various directions in *Frame 310*.

The graphs in Figure 6.10 present the overall detection of corresponding behaviors including the total flow-block (TB), the flow-block detected as normal (NB) and the flow-block detected as abnormal (AbNB) at every time instance. The graphs in Figure 6.10(a) and (b) illustrate the crowd behaviors of sequence shown in Figure 6.8 and Figure 6.9, respectively where the initial frames of the sequence exhibits absolute normal crowd behavior while it tends towards abnormality in gradual and abrupt manner. For instance, in the graph of Figure 6.10(a), after *Frame*

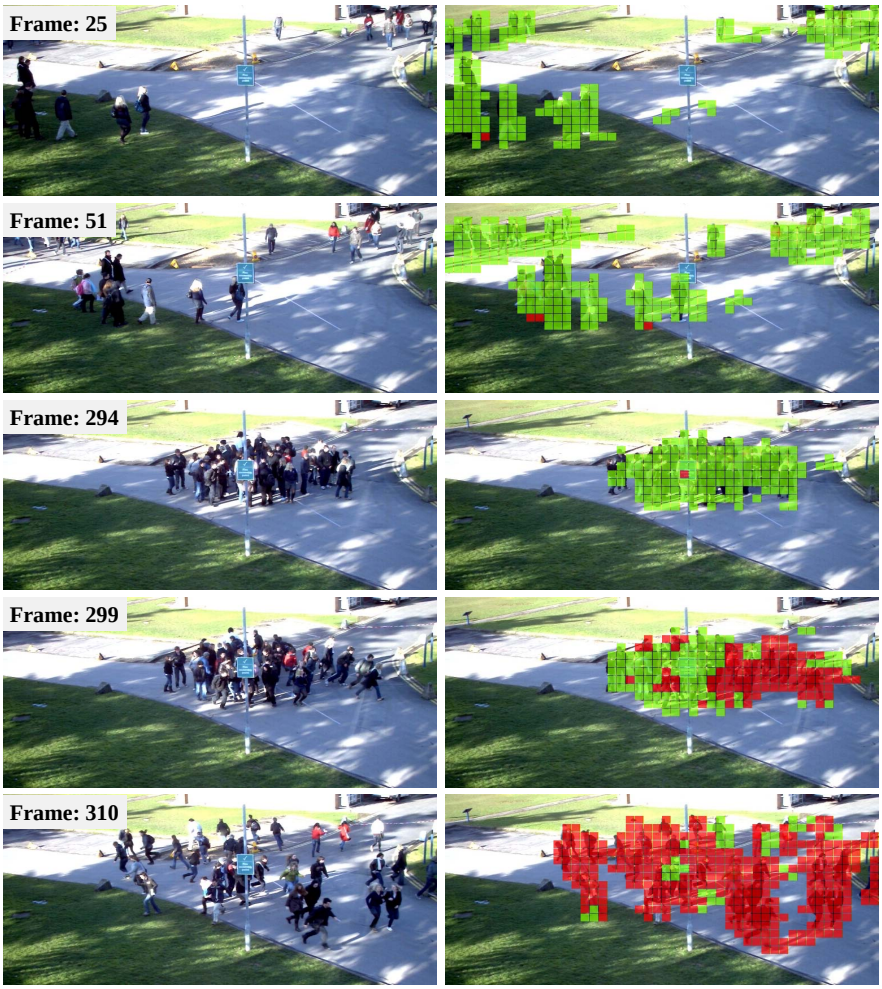


Figure 6.9: shows the detection results on PETS2009 [3] depicting the events of gathering and dispersion. The normal behaviors are indicated by green blocks and abnormal behaviors are marked with red blocks.

56, the abnormality started to appear which is increased monotonically whereas the normal behavior tends to decrease. In contrast, graph in Figure 6.10(b) shows the crowd behaviors where frames till *Frame 300* exhibits absolute normal crowd behavior while it indicates the abrupt abnormality (i.e., dispersion) after *Frame 305*.

Figures 6.11 and 6.12 show the behavior understanding results on the sequences taken from UMN [2] dataset. In the first sequence, shown in the Figure 6.11, people are standing in group and moving around the scene. The corresponding behaviors are detected as normal in *Frame 200*. After some instance of time, a few people start running, randomly in *Frame 284*. As a consequence, people standing in

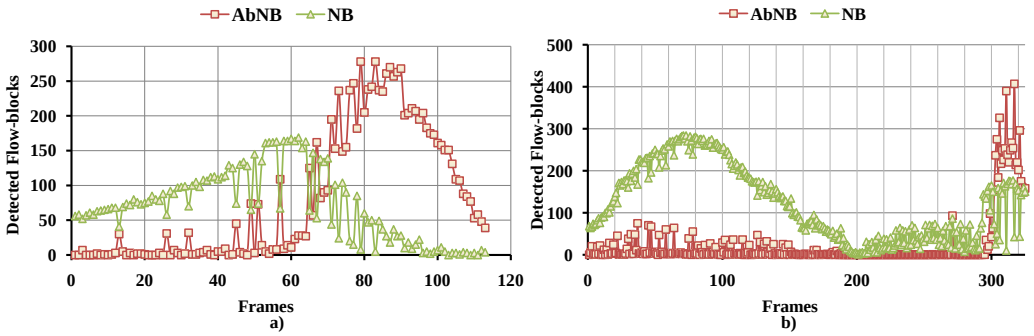


Figure 6.10: shows the detection results on PETS2009 [3] depicting the events of gathering and dispersion. a) and b) present the detected behaviors (i.e., normal flow-block (NB) and abnormal flow-block (AbNB)). The graph presents the corresponding detected behaviors with respective trends in Figure 6.8 and Figure 6.9.

surroundings observed this situations and started running in the similar direction as depicted in *Frame 285*. In this particular case, the people are not pretending to have any common goal but when panic begins in *Frame 305*, all the people respond to the situation and escape from the scene following the similar direction (i.e., towards the right side of the scene).

Figure 6.12 presents a quite challenging scene in terms of people detection and flow computation because the objects are very small in scale. This scene infact mimics the first case but the behaviors of the people are different when they have started running. The objects are standing in the lawn with common pace as shown in *Frame 100* and *Frame 200*. Some people start running aimlessly in *Frame 402* which consequently led the people standing nearby to start running in various directions and resulting in panic as shown in *Frame 415*.

The graphs in Figure 6.13(a) and (b) illustrate the crowd behaviors of sequence shown in Figure 6.11 and Figure 6.12, respectively where the behaviors of objects in the crowd is quite uniform. But, it can be seen that in the graph of Figure 6.13(a), after *Frame 280*, the detection of abnormal flow-block is increased instantaneously whereas the detection of normal flow-block is decreased. Similarly, graph in Figure 6.13(b) shows the crowd behaviors where the frames till *Frame 380* exhibits the normal crowd behaviors while the abnormal behaviors are observed very promptly after *Frame 410*. The results demonstrated here have shown the performance of the proposed approach in detecting the specific and overall behaviors of crowds under different contexts.

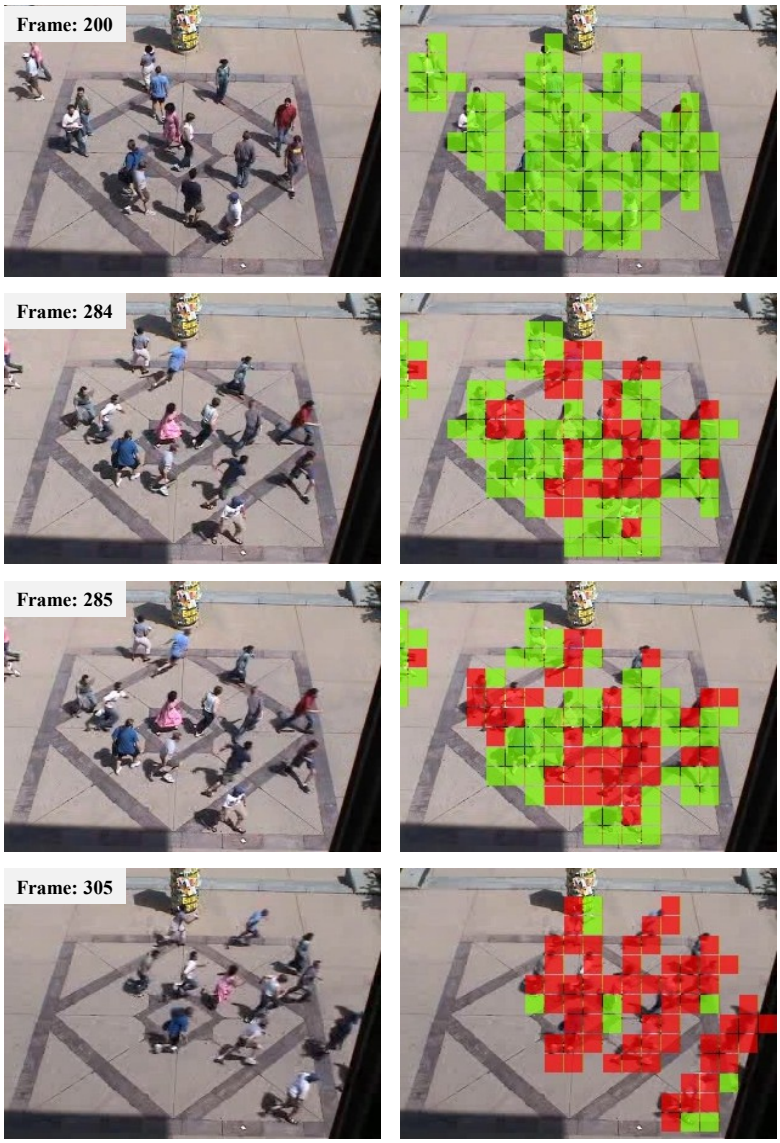


Figure 6.11: shows the detection results on UMN [2] dataset depicting the events of gathering and dispersion. The normal behaviors are indicated by green blocks and abnormal behaviors are marked with red blocks.

6.8.3 Experiments with CRF

In this section, we have presented the result conducted with CRF [12] classification approach over test sequences in Table 6.2. Moreover, the detection of normal and abnormal behaviors are mapped on respective frames for visual representation (see Appendix Figures A.12-A.15). In the next sections, we have described the anomaly detection and quantitative analysis of the results.

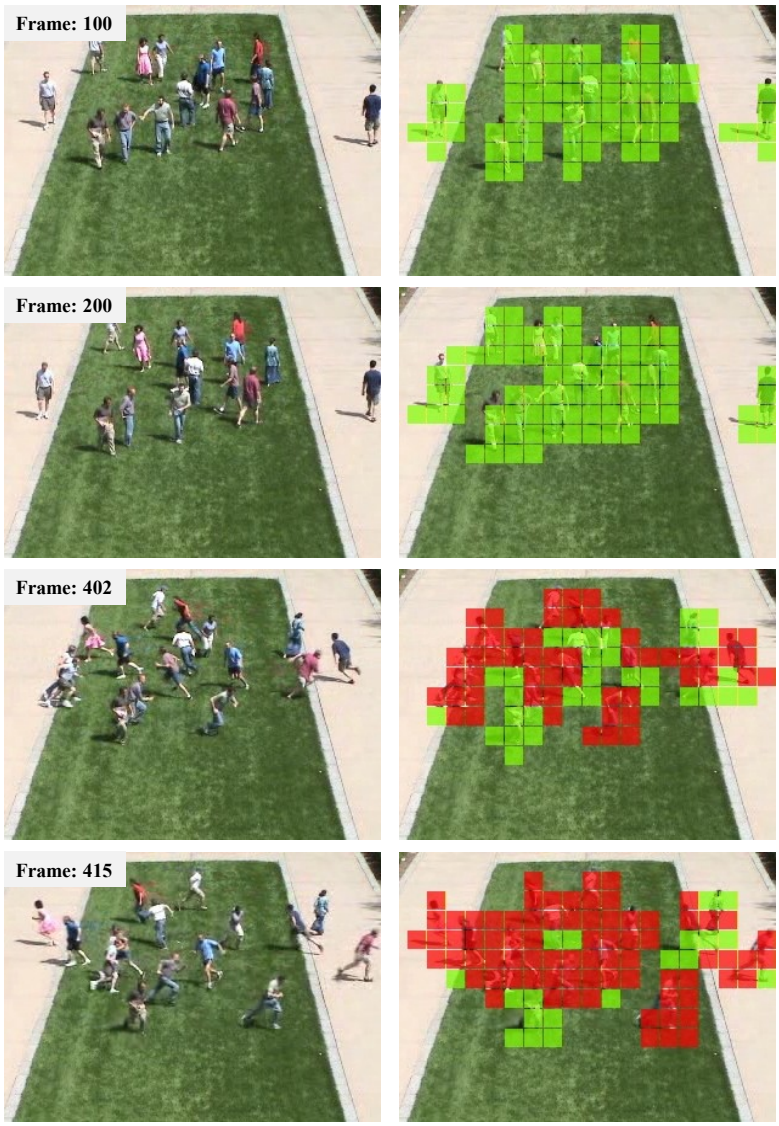


Figure 6.12: shows the detection results on UMN [2] dataset depicting the events of gathering and dispersion. The normal behaviors are indicated by green blocks and abnormal behaviors are marked with red blocks.

Qualitative Analysis

Figures 6.14 and 6.15 show the behavior understanding results on PETS2009 [3].

The first sequence, shown in Figure 6.14, the people are walking with normal pace in the scene at *Frame 25* but after some time instances at *Frame 61*, the leading member of the group starts running. As, the people follows a specific goal so, the people walking nearby also started running at *Frame 72*. The common dynamics of

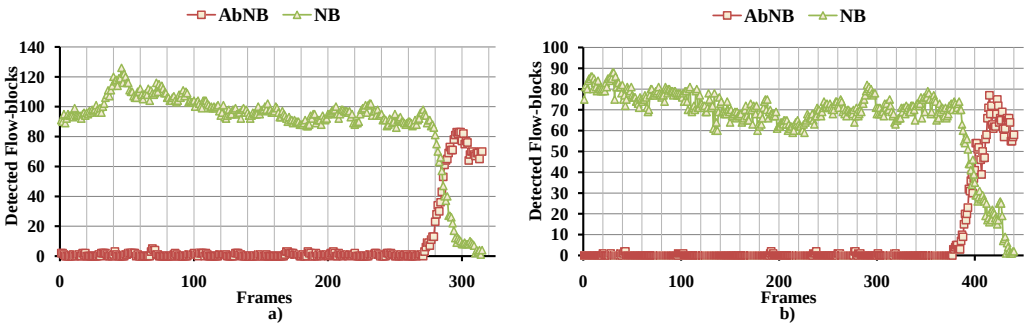


Figure 6.13: shows the detection results on UMN [2] sequences. a) and b) present the detected behaviors (i.e., normal flow-block (NB) and abnormal flow-block (AbNB)). The graph presents the corresponding detected behaviors with respective trends in Figure 6.11 and Figure 6.12.

crowd is now turned into the transition mode where some people are still walking whereas the people at the front are running at *Frame 87*. Later in *Frame 94*, we have seen that all the people started running in the scene.

In the second sequence, shown in Figure 6.15, people are entering with common goal that is to gather in the center of the scene at *Frame 25* and *Frame 51*. The group of people stayed gathered for sometime in *Frame 294*. But, at *Frame 299*, it is shown that suddenly some people starts running at random. As soon as, the other people observed this attitude (i.e., panic), the surrounding people started running too, in various directions in *Frame 310*. The common dynamics of the gathered group is now turned into dispersion (i.e., people are running in various direction).

The graphs in Figure 6.16 present the overall detection of corresponding behaviors at each time instance. The graphs in Figure 6.16(a) and (b) illustrate the crowd behaviors of sequence as shown in Figure 6.14 and Figure 6.15, respectively. The initial frames of both the sequences exhibit absolute normal crowd behavior whereas the abnormal behaviors including running and dispersion is observed in later time instances. For instance, in the graph of Figure 6.10(a), the abnormality started to appear which is increased monotonically in *Frame 56*, whereas the normal behavior tends to decrease in *Frame 80*. In contrast, graph in Figure 6.10(b) the crowd behaviors turned from normal to abnormal behavior after *Frame 310*.

Figures 6.17 and 6.18 show the behavior understanding results on the sequences in the UMN [2] dataset. In Figure 6.17, a number of people are standing in group and moving around the scene in *Frame 200*. Some people, at sudden started running in *Frame 284* and *Frame 285* randomly. As a consequence, people standing

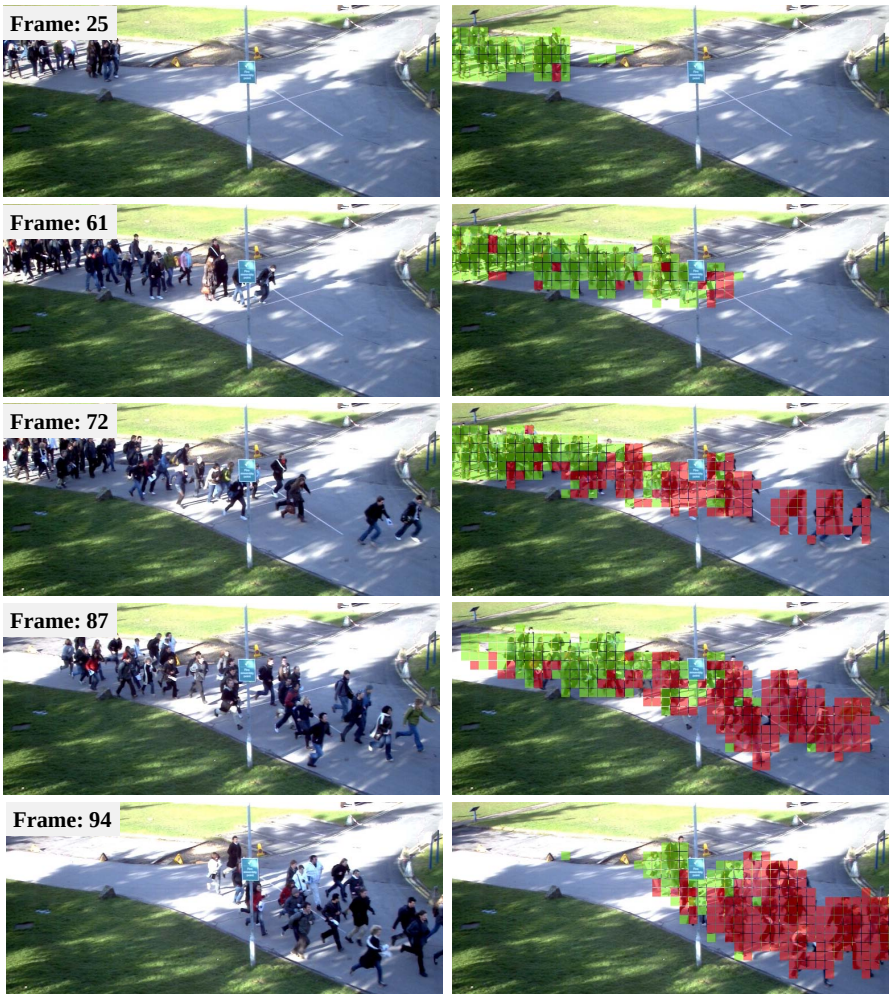


Figure 6.14: shows the detection results on PETS2009 [3]. The normal behaviors are indicated by green blocks and abnormal behaviors are marked with red blocks.

nearby observed this situation and adapt it by start running in the similar direction as depicted in *Frame 305*. In this sequence, it is observed that, when panic begins, all people follow the similar direction to escape (i.e., towards the right side of the scene). The sequence presented in Figure 6.18, people are standing in a lawn and walking in usual manner at *Frame 100* and *Frame 200*. After some instance, couple of people started running aimlessly at *Frame 402* which consequently results in panic. The people standing in surrounding vicinity at *Frame 415* reacted in the similar manner and start running in random direction.

The graphs in Figure 6.19(a) and (b) illustrate the crowd behaviors of sequence as shown in Figure 6.17 and Figure 6.18. In the graph of Figure 6.19(a), after *Frame 280*, the detection of abnormal flow-block is increased instantaneously whereas the

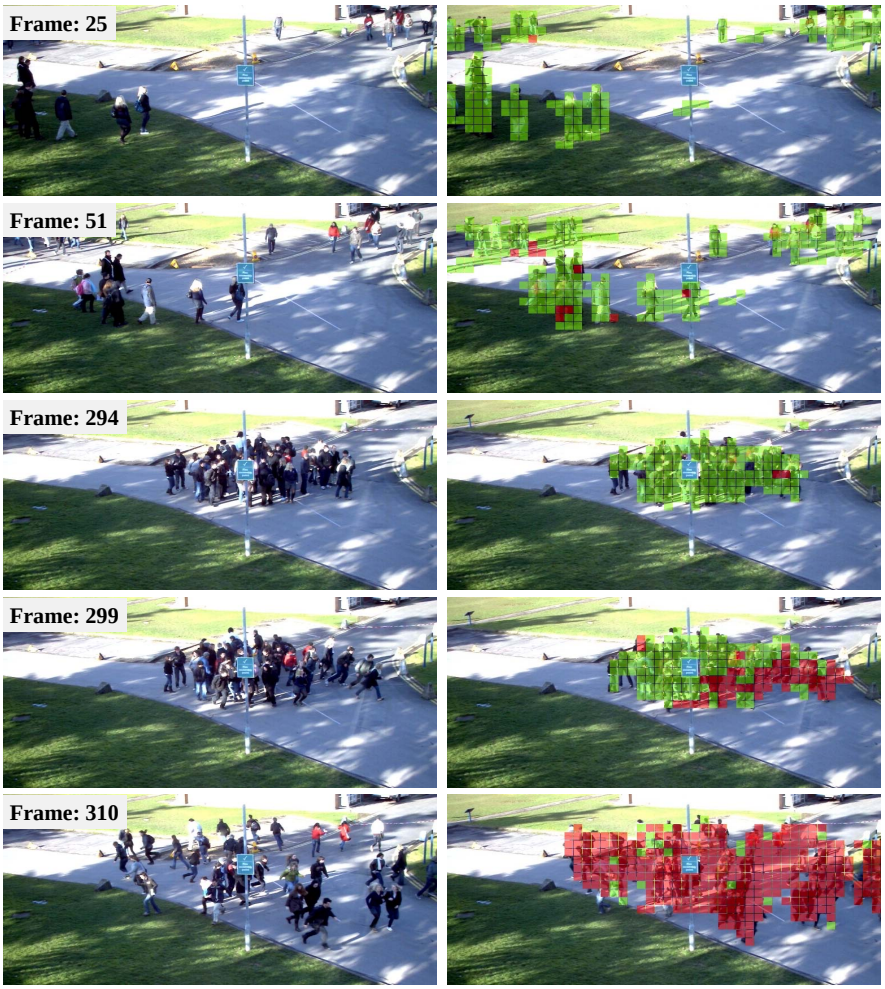


Figure 6.15: shows the detection results on PETS2009 [3]. The normal behaviors are indicated by green blocks and abnormal behaviors are marked with red blocks.

detection of normal flow-block is decreased. Similarly, graph in Figure 6.19(b) shows crowd behaviors where the frames till *Frame 380* exhibits the normal crowd behavior while the abnormal behavior is observed very promptly after *Frame 410*. The results show that the proposed approach is capable of locating the specific and overall crowd behavior in the flow-blocks that are occupied by the crowd.

6.8.4 Evaluation

In this section, we have presented the anomaly detection criteria based on our classification results. Next, we provide quantitative analysis, comparative analysis of classification outcomes, and analysis with state of the art approaches. Further, in

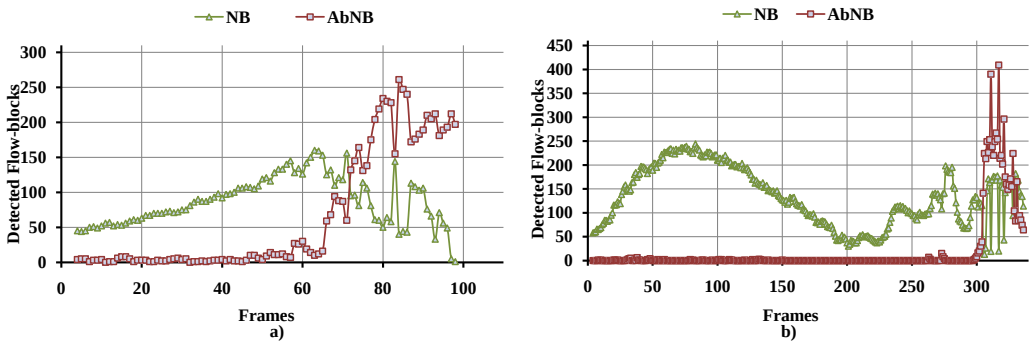


Figure 6.16: shows the detection results on PETS2009 [3] sequence. a) and b) present the detected behaviors (i.e., normal flow-block (NB) and abnormal flow-block (AbNB)). The graph presents the corresponding detected behaviors with respective trends in Figure 6.14 and 6.15.

the quantitative analysis, we also demonstrate the improvement in performance when social entropy is incorporated to handle optical flow noise. Finally, we have underlined the remaining problems which are not addressed in this work.

Anomaly Detection

The definition of anomaly (i.e., abnormality) is specific to the context. We assume that the categorization of a scene as normal and abnormal is somehow fuzzy. Therefore, we have provided both behaviors for the scene analyst to take assistance from the graphical and visual representation. However, for the sake of automization, we have provided a statistical illustration and an option of defining a threshold (i.e., suppose if the abnormal behavior approaches to more than 50%) which is computed as follow:

$$\%Anomaly = \frac{AbNB}{Nb + AbNB} * 100; \tag{6.28}$$

where $\%Anomaly$ defines the total abnormal behavior in the scene which is computed as the ratio of total flow-blocks detected as abnormal relative to the total normal and abnormal classified flow-blocks. In this manner, we can define the status of a crowded scene and analyse the scene behavior at each time instance. Based on our results presented in Sections 6.8.2 and 6.8.3, we have measured the abnormality percentage of each sequence as shown in Figure 6.20.

Quantitative Analysis

We have evaluated our crowd behavior understanding algorithms on the basis of whether they generate correct behaviors corresponding to each flow-block in the

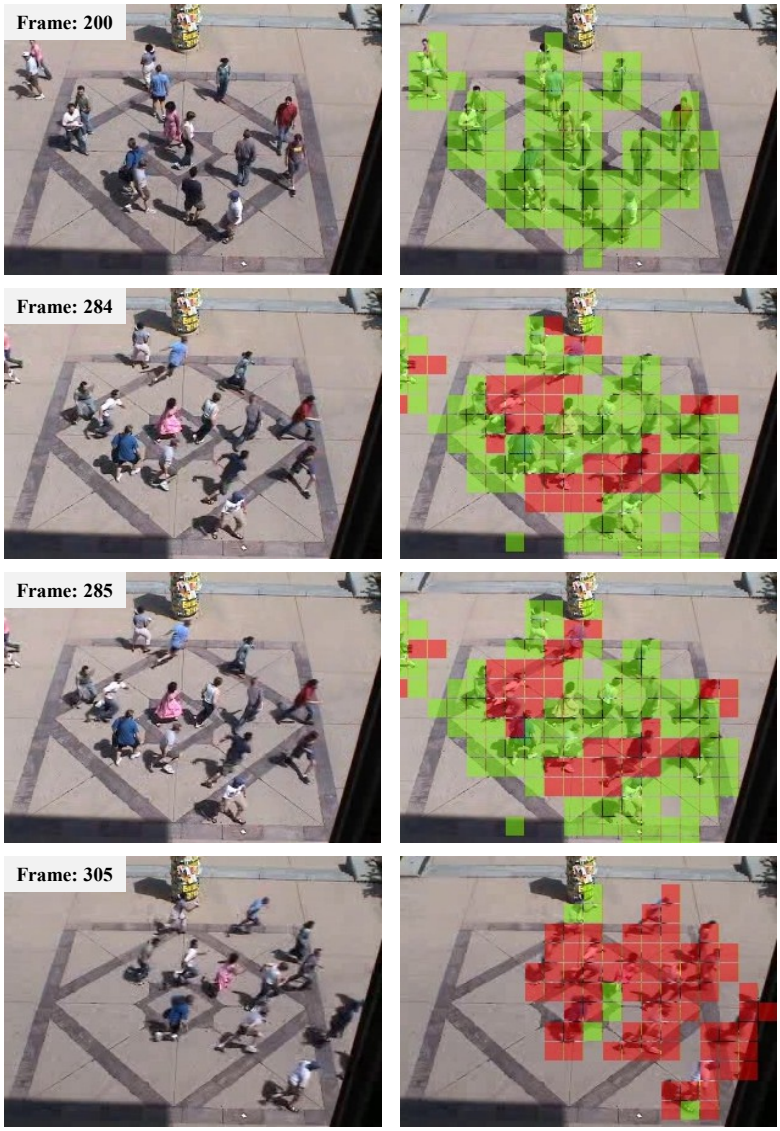


Figure 6.17: shows the detection results on UMN [2] dataset depicting the events of gathering and dispersion. The normal behaviors are indicated by green blocks and abnormal behaviors are marked with red blocks.

sequence. For such evaluation, the first essential requirement is ground truth. For this, we have programmatically create the ground truth[3] for the corresponding behaviors in each flow-block and interpret their respective behaviors. Finally, the performance is evaluated by computing the precision and recall measures. In the

[3]After performing the analysis of normal and abnormal flow ranges on training dataset. We have created a program to label the behavior (e.g., normal and abnormal) of each flow block because, it is quite tedious task to assign labels on each flow-block, manually.

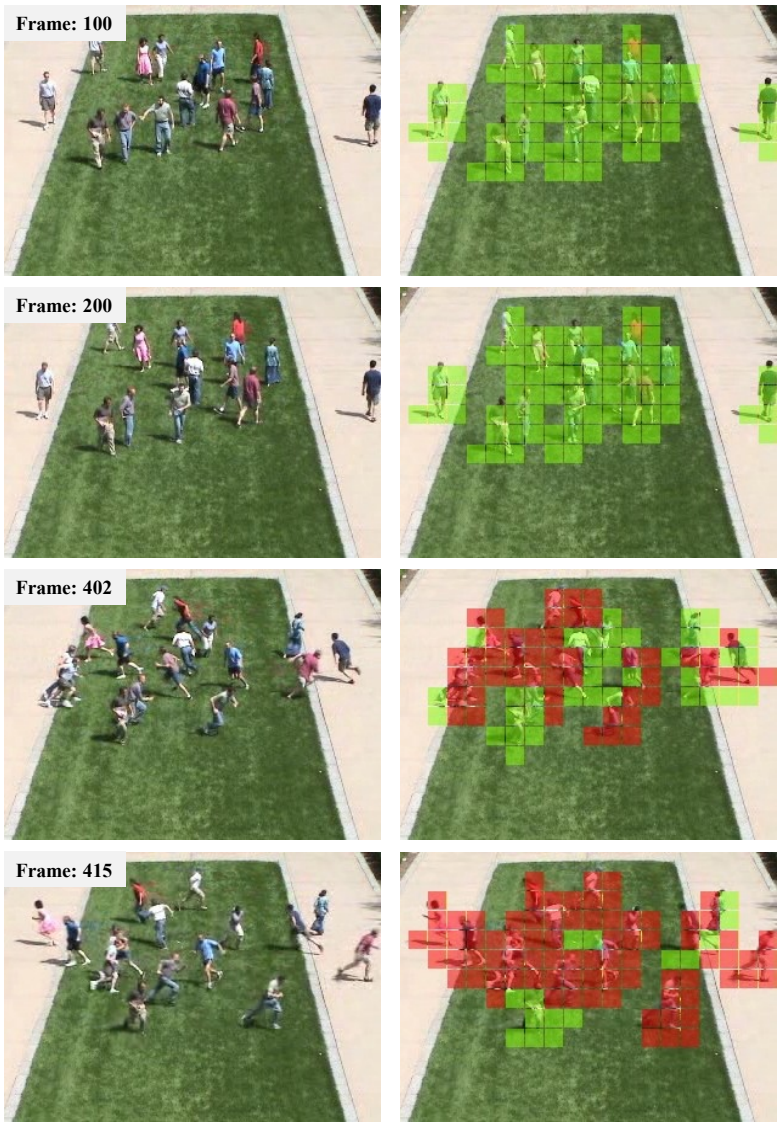


Figure 6.18: shows the detection results on UMN [2] dataset depicting the events of gathering and dispersion. The normal behaviors are indicated by green blocks and abnormal behaviors are marked with red blocks.

context of behavior (i.e., normal abnormal), precision and recall measures are defined as follows:

$$precision\ (pre.) = \frac{Number\ of\ correct\ behaviors}{Number\ of\ established\ behaviors}, \tag{6.29}$$

$$recall\ (rec.) = \frac{Number\ of\ correct\ behaviors}{Number\ of\ actual\ behaviors}, \tag{6.30}$$

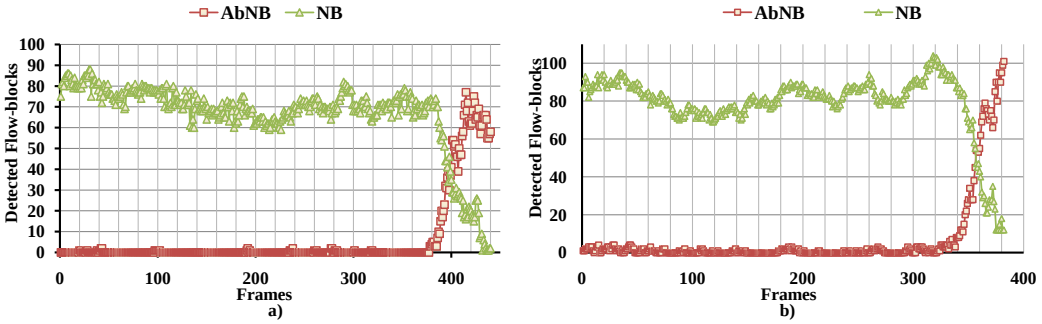


Figure 6.19: shows the detection results on UMN [2] sequence. a) and b) present the detected behaviors (i.e., normal flow-block (NB) and abnormal flow-block (AbNB)). The graph presents the corresponding detected behaviors with respective trends in Figure 6.17 and 6.18.

where actual behaviors denote the behaviors available in the ground truth (i.e., normal and abnormal). Moreover, we have performed two levels of analysis. First, we have presented the classification results without incorporating social entropy. In the second, we have shown the quantitative measurements by incorporating the social entropy.

Table 6.3: Crowd Behavior Detection with SVM Classification on test sequences

	Behavior Detection without SE					Behavior Detection with SE			
	normal		abnormal			normal		abnormal	
Datasets	pre.	rec.	pre.	rec.		pre..	rec.	pre.	rec.
PETS2009 [3]	0.81	0.87	0.71	0.79		0.97	0.98	0.96	0.98
UMN [2]	0.85	0.89	0.70	0.81		0.90	0.91	0.93	0.99
Avg.Values	0.83	0.88	0.70	0.80		0.93	0.94	0.95	0.98

Table 6.3 shows the precision-recall measurements of normal and abnormal behavior classification with SVM [11] for each class. Each column presents the respective precision and recall analysis of each class (i.e., normal and abnormal class). Moreover, the impact of irregular flow on the results is observed due to inter-frame differencing and prominent motion field at objects legs as compared to body and head of the objects as shown in Table 6.3 where flow is modeled without incorporating social entropy (SE). However, the improvements are very vivid in Table 6.3 when social entropy (SE) is incorporated before modeling the observed flow fields.

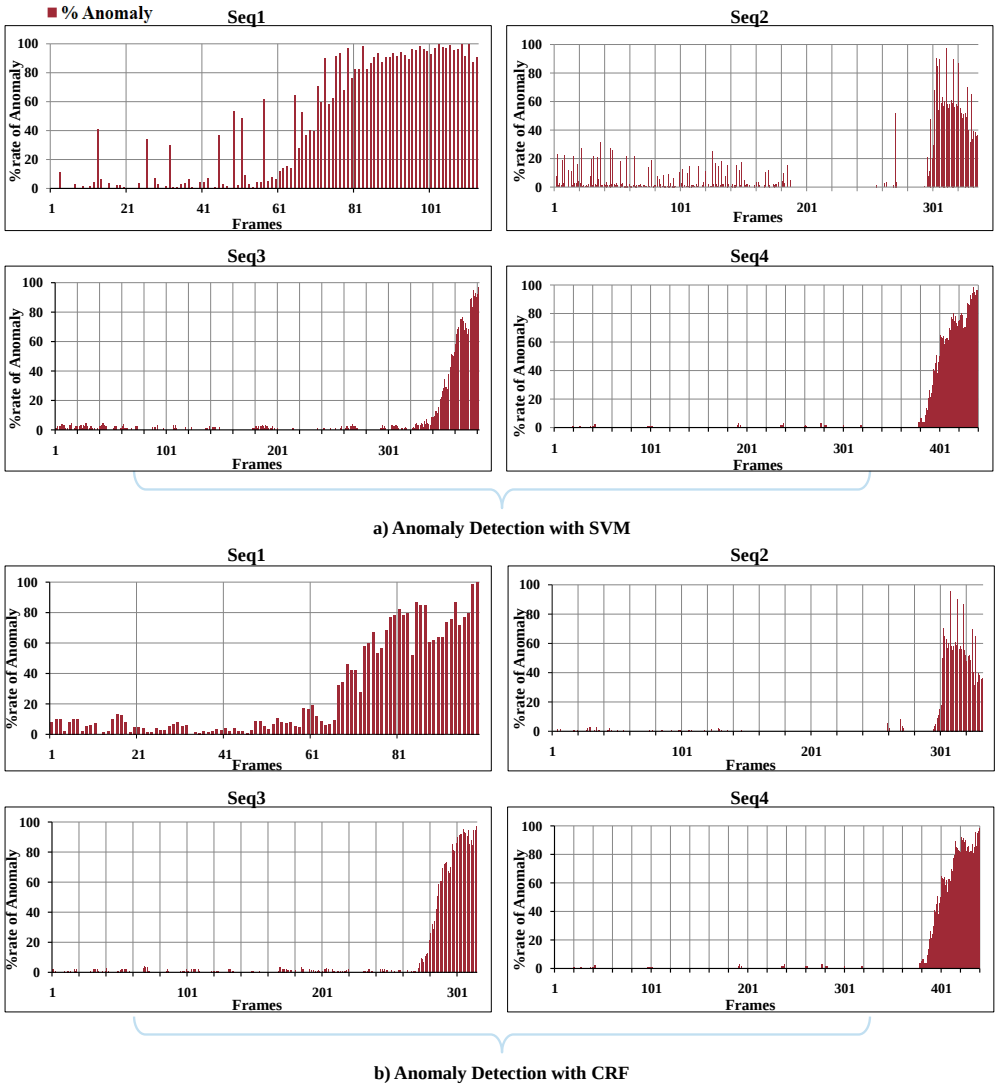


Figure 6.20: graphs demonstrate statistically the relative anomaly observed in sequences from PETS2009 [3] and UMN [2] datasets. a) shows the statistical computation of abnormal behaviors using SVM [11] classification in Seq1 (Figure 6.8), Seq2 (Figure 6.9), Seq3 (Figure 6.11) and Seq4 (Figure 6.12), b) shows the statistical computation of abnormal behaviors using CRF [12] classification in Seq1 (Figure 6.14), Seq2 (Figure 6.15), Seq3 (Figure 6.17), and Seq4 (Figure 6.18).

Table 6.4 shows the precision-recall measurements of normal and abnormal behavior classification with CRF [12] for each behavior class. Each column presents the respective precision and recall analysis of normal and abnormal classes. Table 6.4 shows the classification performance on flow field without incorporating social entropy. It is shown that, the impact of uncertain flow effects the performance in

Table 6.4: Crowd Behavior Detection with CRF Classification on test sequences

	Behavior Detection without SE					Behavior Detection with SE			
	normal		abnormal			normal		abnormal	
Datasets	pre.	rec.	pre.	rec.		pre..	rec.	pre.	rec.
PETS2009 [3]	0.85	0.81	0.73	0.79		0.97	0.98	0.96	0.98
UMN [2]	0.83	0.79	0.70	0.81		0.99	0.98	0.96	0.97
Avg.Values	0.84	0.82	0.72	0.80		0.98	0.98	0.96	0.97

Table 6.5: Comparative analysis with state of the art approaches for PETS2009 [3] and UMN [2] datasets

Methods	Results(%)
Mehran et al.[59]	96
Chen et al.[93]	81
Our Method	97.3

overall precision and recall. However, the improvements is observed in Table 6.4 when social entropy is incorporated before modeling the observed flow fields.

Comparative Analysis

In this section, we have performed a two level comparative analysis. In the first, we have discussed the performance of SVM [11] and CRF [12] classifiers as shown in Table 6.3 and 6.4 and it is observed that CRF has superior performance than SVM [11]. In the second, to analyze the performance of our proposed approach in detecting the crowd dynamics effectively, we have a made comparative analysis from two recent proposed techniques [59] [93]. In the first approach, the computed social forces are modeled with LDA whereas in the second approach, SVM [11] is used to classify the behaviors. It can be seen in Table 6.5, the performance of our method is promising and achieve higher detection rates in the localization of the crowd behaviors when compared with related approaches.

Remaining Problems

In our proposed approach, we have aimed to address the issues of crowd behavior analysis and achieve our defined objectives. However, the proposed approach has certain limitations which can be addressed in our future research.

- Tracking in Crowds: In the proposed approach, we are not tracking individuals because the objects are highly anticipated which results in heavy occlusions. It is possible to employ higher level knowledge and model pedestrian behavior model (i.e., computation flow dynamics) into the tracking algorithm based on the strong assumptions about the pedestrian behaviors.

- Detection of Individual Behaviors: Human forming the crowd has complex physics and self-evolving nature. It is very challenging to detect specific individual and examines the corresponding behaviors because the computed features, such as interest points, localized heads, and specific human classifiers become unreliable. Therefore, currently, we have aimed to localize the crowd behaviors in a scene at the global and specific level.

- High Level Events: The definition of crowd behaviors is very context specific problem. The broad level description includes normal and abnormal behaviors. However, these event can be further categorized into many high level event, such as recognition of dispersion, falling down of persons, and detection of fights. However, these further interpretation can be built on top of proposed approach by employing the strong assumptions according to the social behaviors of individuals and the contexts of application.

6.8.5 Context of Use and Applicability

The proposed framework is aimed to perform analysis of behaviors for crowded scenes such as whether the state of the crowd is normal or abnormal. While researching on this active research idea, we keep the certain contexts of application domain when typical surveillance systems limit. For instance, the way in which crowds move is of central concern for security officials, civilian authorities, and disaster relief agencies to monitor the situations and manage the emergency situations. Moreover, the scenarios such as individual passing through a combat zone to escape conflict, and persons fleeing natural disasters, such as earthquake and fire can be monitored by incorporating the contextual information which will tune the analysis according to specific application. Other domains of application can be explored by incorporating the contextual information and human assistance to construct this proposed framework for more practical real scenarios.

6.9 Discussion and Conclusion

In this chapter, we have aimed to investigate crowd behaviors and detect the trends of anomaly observed during the course of time. To achieve this task, we have proposed a top-to-down framework which begins with low-level analysis (i.e., foreground detection). The sequence is sectioned into video segments and concept of flow-block is introduced where the computed optical flow is treated at flow-block level. Prior to model the observed flow cloud data, optical flow uncertainty is handled by employing the concept of social entropy at each flow-block level whereas the evaluation criteria are learned from the training sequences. Next, the key objective of computing flow features are achieved by parameterizing with mixture of Gaussians and flow features (i.e., flow patterns). These flow features are mapped as two class problem and classified as normal and abnormal using SVM [11] and CRF [12], respectively. The results of our method indicates that the proposed approach is effective in detection and localization of specific and overall behaviors in the crowd. The presented results show promising performance and outperforms when compared to the related works.

CHAPTER 7

Conclusion and Future Directions

In this dissertation, the main theme is to understand object behaviors in the scene depicting both non-crowded and crowded situations. Typical examples of the non-crowded scenes include: train station, subways, and foyer, whereas the crowded scenes include: sporting events, religious festivals, and shopping malls. We have provided a detailed analysis and underlined the related issues in *Chapter 2* according to our adapted strategy of research. *Chapter 3* begins with segmentation (i.e., low level analysis) and eliminates the need for global level analysis that requires high computational efforts. This is achieved by developing a weighted integrated approach which employs both adaptive background mixture model approach and approximated median filter approach to detect the foreground under diversified situations. This segmented information is then used to extract visual features for both non-crowded and crowded scenes in *Chapter 4*. In the non-crowded scene, we have used the color as a fundamental feature and is then treated in the form of ellipse histogram and CSC [4] approach for the detected objects. In contrast, the motion is used as a fundamental feature for crowded scenes.

Next, the segmented information and visual features are employed to develop a tracking and behavior understanding framework in *Chapter 5* that is used to track and understand objects movement behaviors within the scene. For this purpose, we have proposed a new methodology that integrate the quantitative (i.e., statistical modeling) and qualitative approaches with the motivation to address the tracking problem by axiomatizing and reasoning the human-tracking abilities. The results are demonstrated on three benchmark datasets containing conflicted situations. Finally, in *Chapter 6*, we have proposed crowd behavior understanding and anomaly detection algorithms that are aimed to analyze the behaviors of objects in crowded scenes. The proposed algorithm used the optical flow as a main feature and is modeled with mixture of Gaussians to generate flow patterns. These flow patterns are classified with SVM [11] and CRF [12] to detect any changes in crowd behavior, thus enabling localization of normal and abnormal events or behaviors within crowd.

7.1 Summary of Contributions

In this section, we have provided a summary of contributions in this dissertation.

- Segmentation and Feature Computation
  - ◇ Improvement in the segmentation approach by conditional weighted integration of two approaches (i.e., adaptive background mixture model and approximated median filter approach).
  - ◇ Introduction of the idea of using CSC approach and ellipse histogram as object features in non-crowded scenes.
  - ◇ Representation and modeling of optical flow (i.e., flow cloud data) by employing of mixture of Gaussians to obtain crowd flow features.
- Tracking and Behavior Understanding in Non-crowded Scenes
  - ◇ A framework is proposed for understanding and tracking the objects in non-crowded scenes containing objects moving in the complex manner.
  - ◇ Introduction of a new concept to integrate statistical approach (quantitative) with cognitive modeling (qualitative) for the vision community to address the problems of object tracking particularly during conflicted situations (e.g., occlusion or split among the objects).
  - ◇ Fusion of computed features (i.e., CSC color-patches and ellipse histogram) with Bayesian inference, named as Bayesian Matching Weight method to compute the correspondence weights in consecutive time instances.
- Behavior Understanding and Anomaly Detection in Crowded Scene
  - ◇ A framework for understanding the crowd behaviors and anomaly detection is proposed.
  - ◇ Introduction of the idea of computing global flow and treating at local level by forming flow-blocks.
  - ◇ Handling optical flow uncertainty by incorporating the social entropy approach.

- ◇ Behavior classification is addressed as binary class problem and classification approaches are employed to localize the corresponding behaviors. Besides, the anomaly is computed at frame level that allows us to use the proposed approach as an application of the abnormal event detection in crowds.

7.2 Future Directions

The work described in this research can be improved and extended in a number of directions for both non-crowded and crowded scenes. There are some evident enhancements which can improve the performance and applicability. Some of these ideas are described as follows:

Non-Crowded Scenes

There are many possible directions which include the improvement of the developed algorithms and building new approaches on top of it. The performance of matching algorithm used in tracking framework can be greatly improved at the detection and visual feature level. The robust segmentation provides a good foundation to compute the features that best discriminate the objects in the scene over time. The matching methods proposed in this dissertation use a BMW method to combine computed ellipse histogram and CSC color-patches. However, the same feature (i.e., color) may not be the most discriminative for an object over time. One possible direction is to consider different features such as motion and textures. Then, fusion of these features can be performed by the classification algorithm such as SVM to obtain the best matching performance. Given a large set of features, SVM classifier can be trained for each feature treating as an individual class. So, the classifier discovers weighted combination of classification outcomes.

In this work, we have provided the logical interpretation of object motion behaviors only. Further, high level logical modeling can be incorporated to infer activities, such as sitting or lying in the scene. Another possible future work is to detect the object's belonging such as bags. For this purpose, specific approaches, such as the histogram of gradients can be used for object detection and Adaboost for bags detection. These detection results are then mapped logically in owner-belonging relationships which can be tracked and un-attended belonging are detected at the same time.

Crowded Scenes

In this dissertation, we have performed crowd behavior analysis to understand objects behavior and to detect anomaly. However, we have not tried the tracking of objects in the crowded scene due to various limitations and constraints (e.g., heavy occlusions and complex interactions among individuals). So, the work can be extended to track the specific targeted objects in the crowded scenes. For this purpose, the target-specific particle filter system can be employed. It will add another level of analysis related to specific object behaviors in the crowds, and the tracking axioms proposed for non-crowded scenes can be derived for the behaviors in crowded scenes.

The behaviors which we have considered are categorized as normal or abnormal. However, a more refined analysis of crowd behavior can be performed by incorporating crowd psychological research. For instance, the common types of crowd behaviors that can be detected includes: lane formation, bottlenecks, intersections, dispersions, and panic effects. This can be achieved by incorporating the context semantics on top of the current behavior understanding algorithm. Moreover, the temporal history of the classified outcome of flow-blocks for each crowd behavior can be taken into account and mapped on the models of the crowd psychological conditions. In this manner, we will be able to generate a representation of the crowded scene that is easier for human operators to understand and interpret.

APPENDIX A

Appendix

A.1 HSV Color Space

A widely used color space to represent the color information of an image is RGB but HSV is preferred as the color space because it represents human perception of colors in better way. HSV color space comprises of Hue (H) as color component, Saturation (S) as intrinsic color and describes purity of color, and Value (V) is the measure of brightness component. The HSV color model is simply computed by using standard formulation as:

$$H = \begin{cases} 0 & if\ max = min \\ 60^o \times \frac{G-B}{max-min} + 0^\circ & if\ max = R and G \geq B \\ 60^o \times \frac{G-B}{max-min} + 360^\circ & if\ max = R and G < B \\ 60^o \times \frac{B-R}{max-min} + 120^\circ & if\ max = G \\ 60^o \times \frac{R-G}{max-min} + 240^\circ & if\ max = B \end{cases} \tag{A.1}$$

$$S = \begin{cases} 0 & if max = 0 \\ \frac{max-min}{max} & otherwise \end{cases} \tag{A.2}$$

$$V = max \tag{A.3}$$

A.2 Histogram Computation

In this section, we have presented the fundamental formulation of computing color histogram for the images. The colors of image are mapped into a discrete color space $\mathcal{R}$ which is divided into N intervals. In practice, these intervals are regularly spaced, but it can also be irregular. Given a HSV color space $\mathcal{R}$ which is divided into N intervals as follows:

$$\mathcal{R}^n \subset \mathcal{R}, \tag{A.4}$$

with

$$\bigcup_{n=0}^{N-1} = \mathcal{R}, \tag{A.5}$$

The probabilities of given observations are computed by counting the number of observations that fall in a specific interval. As an example, we assume that L^n of M observations fall into $\mathcal{R}^n$ histogram interval, we get the following estimated probability P_n:

$$P_n := P(s \in \mathcal{R}^n) = \frac{L^n}{M}; \tag{A.6}$$

As, our color space is divided into a regular interval of volume Q. Therefore, the probability density of an observation $s \in \mathcal{R}^n$ can be modeled using piecewise constant function, given as follows:

$$p(s) = \frac{L^n}{MQ}, \quad s \in \mathcal{R}^n, \quad n = 0, \ldots, N-1; \tag{A.7}$$

The Equation A.6 can be written as follows:

$$P_n = \int_{\mathcal{R}^n} p(s)ds = p(s\,|(s \in \mathcal{R}^n))Q = \frac{L^n}{M}; \tag{A.8}$$

The histogram intervals are indexed by numbers and called number of bins (i.e., intervals) where the selection of intervals is empirical[1].

A.3 Color Structure Code Approach

The CSC algorithm is an improved region growing method used to segment object corresponding to its homogeneous regions. The approach [4] follows a parallel hierarchical region growing method on a special hexagonal topology; therefore the choice of the starting point and the order of processing are not required. The hierarchical topology is constituted by islands which form at different levels. An island consists of seven pixels, one at the center with six equidistant neighbors. A partitioned image contains islands which are overlapping in a manner where each second pixel of each second row is a center of an island of level 0. Likewise, an island of level $(n+1)$ is built up consisting of seven overlapping islands of level n. This process is repeated until one island covers the whole image and the number of islands decreases by a factor of 4 from one level to another level.

In practice, the hexagonal topology leads to some difficulties due to orthogonal lattice produced by cameras. So it is assumed that all the pixels are presented in a hexagonal arrangement which is achieved by simulating a conventional orthogonal

[1] Based on the analysis presented in [98], we have selected 45 intervals for our histograms.

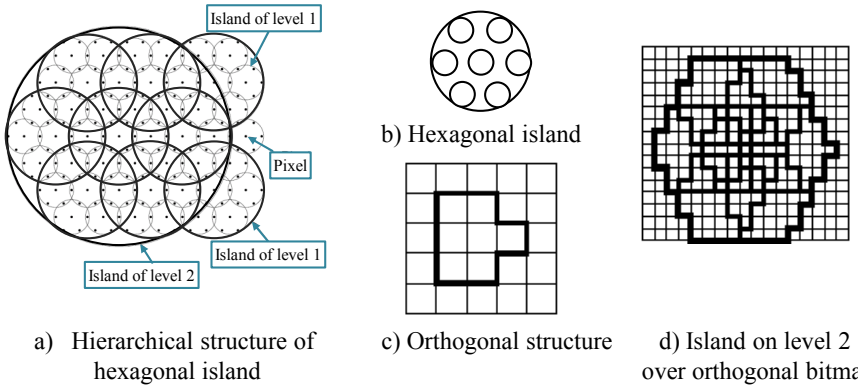


Figure A.1: shows the CSC Hierarchy [4]. a) describes the hierarchical structure of hexagonal architecture, b) and c) presents the hexagonal island and orthogonal structure, d) indicates the islands of level 0 upon the orthogonal lattice.

images as shown in Figure A.1. The generation of the CSC segments operates essentially in three phases:

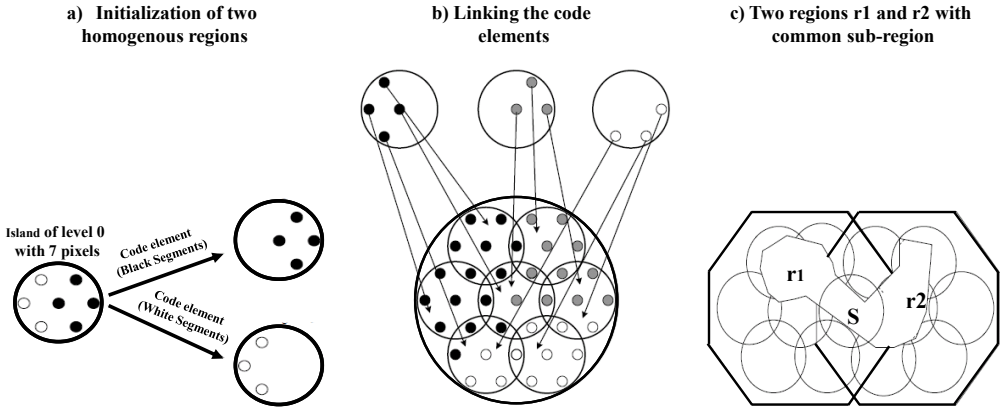


Figure A.2: demonstrates the phases of CSC methods: a) shows initialization phase of two homogeneous regions (i.e. black and white), b) presents linking mechanism of homogeneous regions, and c) demonstrates splitting phase based on selected parameters and final segments are formed after satisfying the given criterion.

Initialization

In the initialization phase, image homogeneous color regions within level 0 of island of seven pixels are divided up and mapped to initial code elements. These code-element comprises of neighboring pixels within level 0 islands whose colors are similar (i.e. their mutual color distance is under a certain threshold). A

code-element is a data structure containing the information about the shape and the mean-color of a region in an island. For example, in Figure A.2(a), there are two code-elements in this island due to two different colored regions. Next, in the linking phase, connected color segments grow hierarchically by checking these small color patches.

Linking

In linking phase, the color-elements of level n grow in a hierarchical manner to new code-element of level $n+1$ in the seven neighbored overlapping islands of the hexagonal island structure. The linking of code-elements within one island is similar to initialization phase where instead of linking single pixels, regions are linked. Similarly, the criteria for linking code-elements are based on their similarity in color whereas two code-elements are connected if they share a common sub-region in their common sub-island as shown in Figure A.2(b). The linking operations are repeated for all islands on every level, starting from level 1 and ending on the topmost level which covers the whole image.

Splitting

In splitting phase, a chain of changing colors segments which are observed during linking are split again into segments representing homogeneous color. In this manner, the problem of linking different colored regions by local region growing techniques is resolved. During splitting, additional color similarity is checked between connected code-elements at every linking level. The decision of connectivity is based on their color distance which should satisfy the empirical selected threshold, although they are connected by a chain of similar color pixels. For example in Figure A.2(c), the two regions $r1$ and $r2$ are not linked anymore because the color distance is too high. Moreover, this mechanism enables the smooth transition from one region to another at global level.

The CSC method suggested in [4] is employed on our test datasets in Figure A.3 with the goal of revealing the distinct ability of detected objects and segments objects according to its color structures. For instance, in Figure A.3(b-c), object itself contains different color shades relative to dress, bags, and hat.

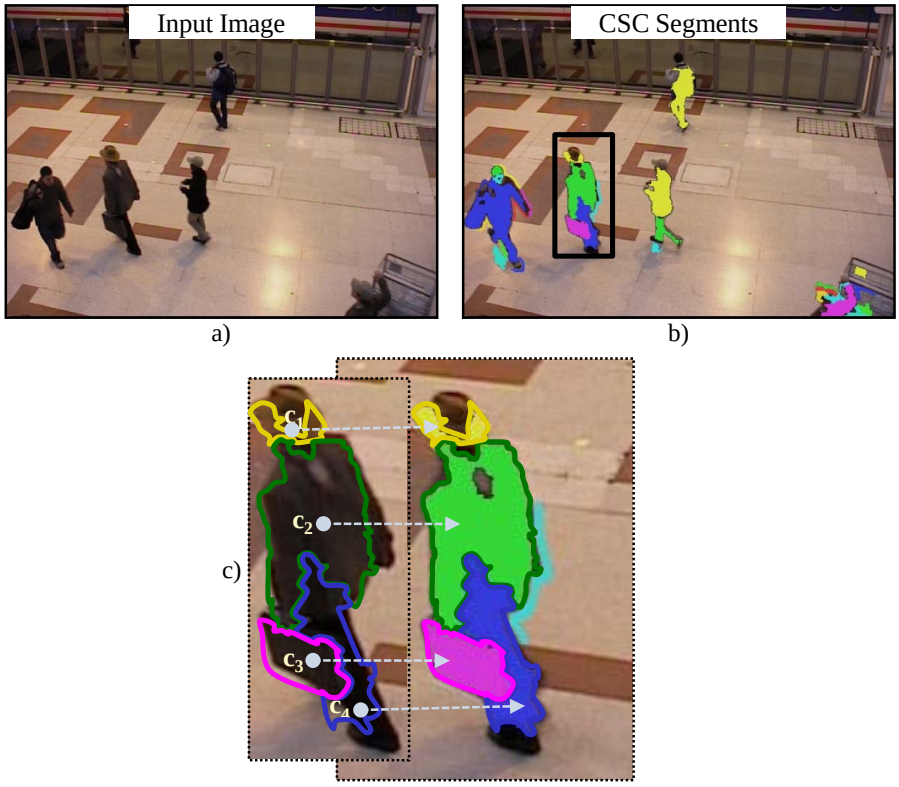


Figure A.3: a) is the input image of PETS2006 [1] sequence, b) shows results of CSC [4] approach on a sample frame of PETS2006 [1] sequence. c) indicates CSC color-patches which are encoded with different colors codes for better visibility. The zoomed object is comprised of four major color: 1) hat, 2) coat, 3) trouser, and 4) briefcase, in the actual image. Each color-patch is linked through arrows with its corresponding color.

A.4 Tracking and Behavior Understanding in Non-crowded Scenes

In this appendix, we have presented additional results in Figure A.4-A.6 of our proposed approach for tracking and behavior understanding of objects in non-crowded scenes. In Figure A.4(a) shows the input sequences from PETS2006 [1] dataset. Figure A.4(b) shows the segmentation results in Section 3.1 on the test sequence from PETS2006 [1] dataset. Figure A.4(c) shows the visual mapping of bounding box in Section 2.2.2 on the test sequence from PETS2006 [1] dataset. Figure A.4(d) shows visual mapping of contours in Section 2.2.2 on the test sequence from PETS2006 [1] dataset. Figure A.4(e) shows visual mapping of computed ellipse in Section 2.2.2 on the test sequence from PETS2006 [1] dataset.

Figure A.4(f) shows results of CSC approach in Section 4.1.2 on the test sequence from PETS2006 [1] dataset. Figure A.4(f) shows results of quantitative and qualitative approach to infer object's behavioral states and manage unique identities over time in Section 5.4 on the test sequence from PETS2006 [1] dataset. Figure A.4(h) shows the results of Kalman filter-based tracker in Section 5.5 for object localization on the test sequence from PETS2006 [1] datasets.

A.5 Behavior Detection and Understanding in Crowded Scenes

In this appendix, we have shown the additional results Figure A.10-A.11 of our proposed approach for crowd behavior detection. In Figure A.10(a) shows input sequences from PETS2009 [3] dataset. Figure A.4(b) shows computed optical flow on foreground region in Section 6.4 on the test sequence from PETS2009 [3] dataset. Figure A.10(c) shows visual mapping of motion vectors based on optical flow in Section 6.4 on the test sequence from PETS2009 [3] dataset. Figure A.10(d) shows behavior detection with CRF classification approach in Section 6.7 on the test sequence from PETS2009 [1] dataset. Figure A.10(e) shows behavior detection with SVM classification approach in Section 6.6 on the test sequence from PETS2009 [1] dataset.

Figure A.4: shows results on PETS2006 [1]. a) input sequence, b) the extracted objects from segmentation, c) the computed bounding region over the detected objects, d) contour of the objects, e) ellipse around the object which are used to compute the ellipse histogram, f) CSC color-patches of each detected object, g) and h) demonstrate the results tracking and behavior understanding approach.

Figure A.5: shows results on PETS2006 [1]. a) input sequence, b) the extracted objects from segmentation, c) the computed bounding region over the detected objects, d) contour of the objects, e) ellipse around the object which are used to compute the ellipse histogram, f) CSC color-patches of each detected object, g) demonstrates the results tracking and behavior understanding approach.

Figure A.6: shows results on PETS2009 [3]. a) input sequence, b) the extracted objects from segmentation, c) the computed bounding region over the detected objects, d) contour of the objects, e) ellipse around the object which are used to compute the ellipse histogram, f) CSC color-patches of each detected object, g) and h) demonstrate the results tracking and behavior understanding approach.

Figure A.7: shows results on IESK dataset. a) input sequence, b) the extracted objects from segmentation, c) the computed bounding region over the detected objects, d) contour of the objects, e) ellipse around the object which are used to compute the ellipse histogram, f) CSC color-patches of each detected object, g) demonstrates the results tracking and behavior understanding approach.

Figure A.8: shows results on PETS2006 [1] dataset. a) the extracted objects from segmentation, b) the computed bounding region over the detected objects, c) ellipse around the object which are used to compute the ellipse histogram and d) shows the CSC color-patches of each detected object.

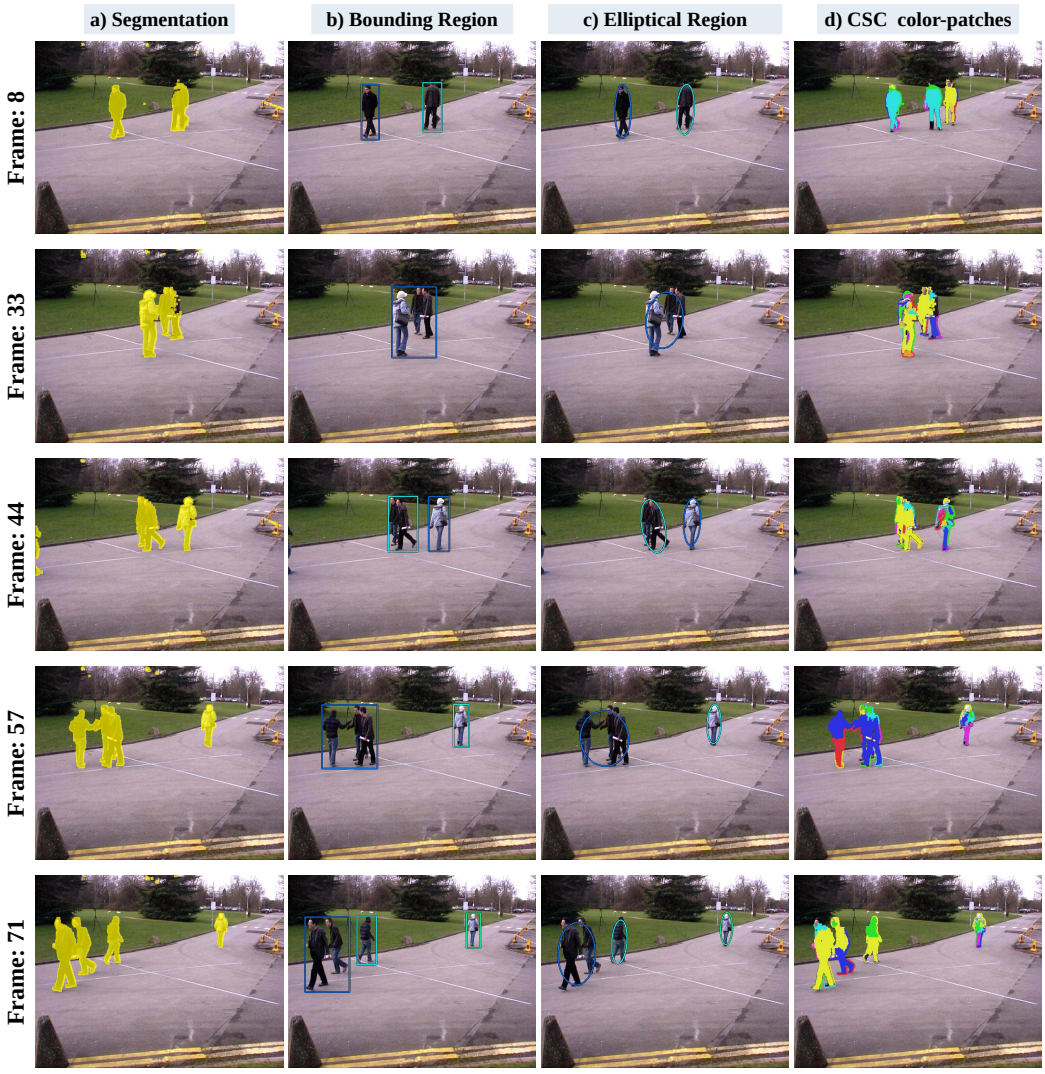


Figure A.9: shows results on PETS2009 [3] dataset. a) the extracted objects from segmentation, b) the computed bounding region over the detected objects, c) ellipse around the object which are used to compute the ellipse histogram and d) shows the CSC color-patches of each detected object.

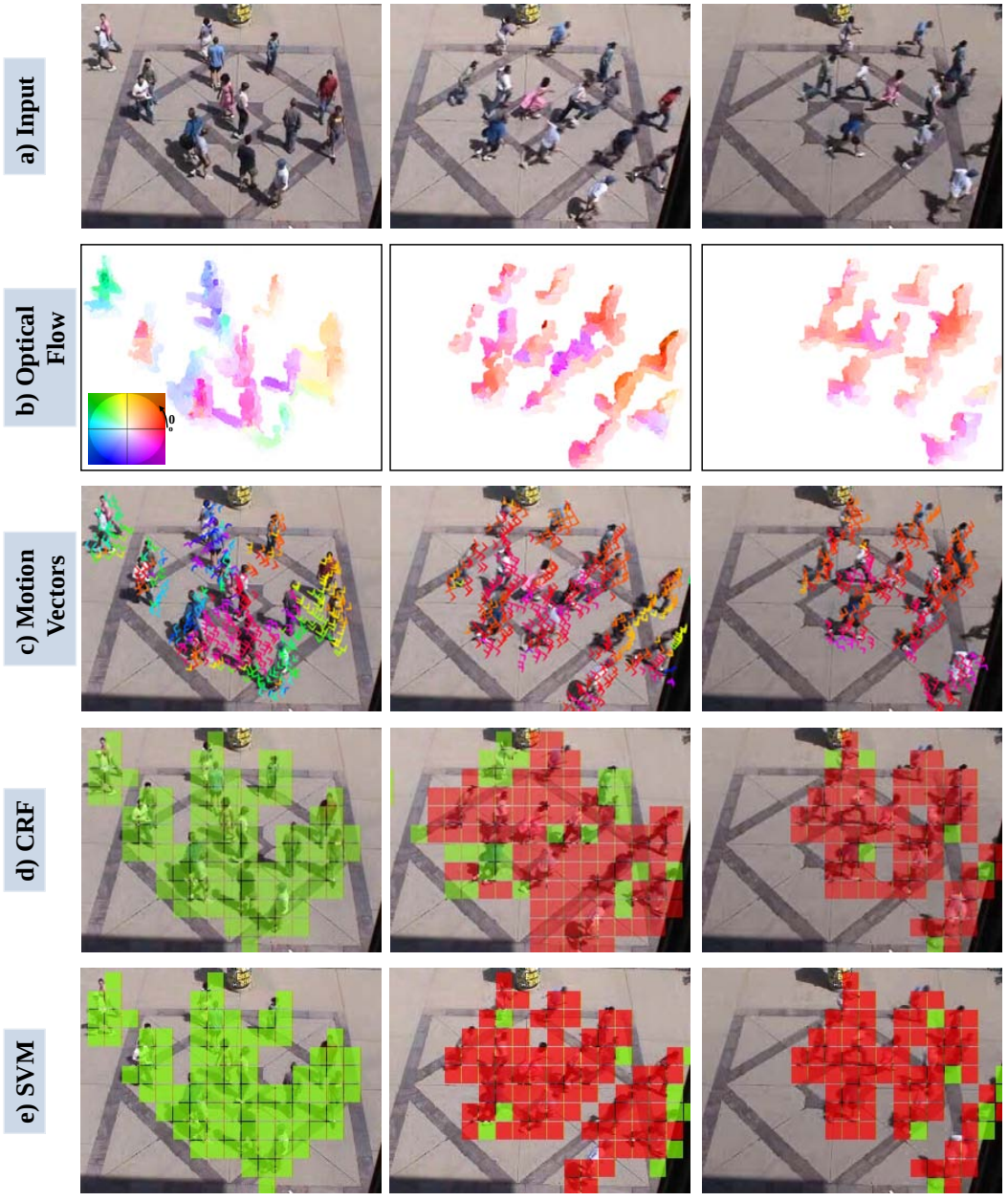


Figure A.10: shows results on UMN [2] dataset. a) input sequence, b) the optical flow computed on segmented region which is presented by using the color encoding scheme [102], c) shows the flow vectors, d) demonstrates the classification of flow-blocks with CRF [12], and e) presents the classification of flow-blocks with SVM [11].

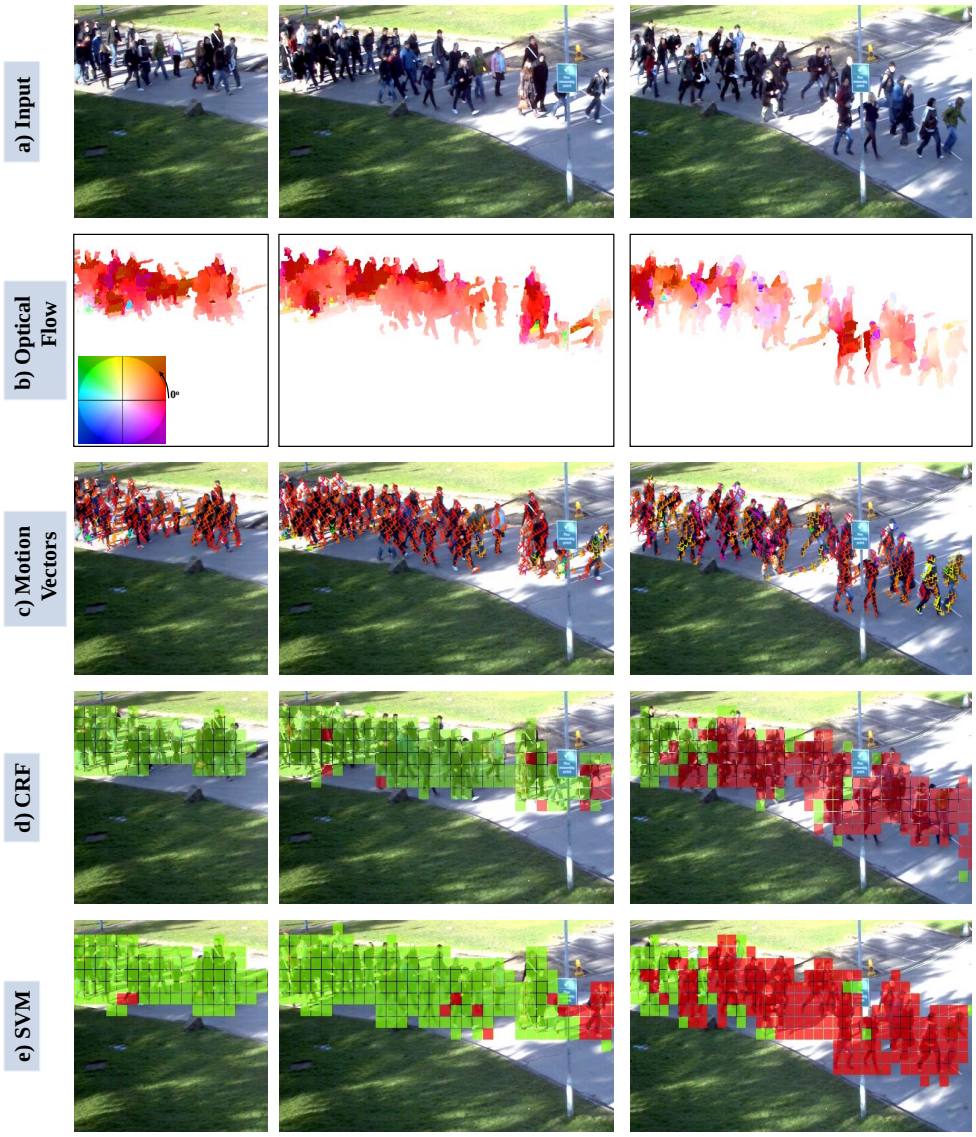


Figure A.11: shows results on PETS2009 [3] datase. a) input sequence, b) the optical flow computed on segmented region which is presented by using the color encoding scheme [102], c) shows the flow vectors, d) demonstrates the classification of flow-blocks with CRF [12], and e) presents the classification of flow-blocks with SVM [11].

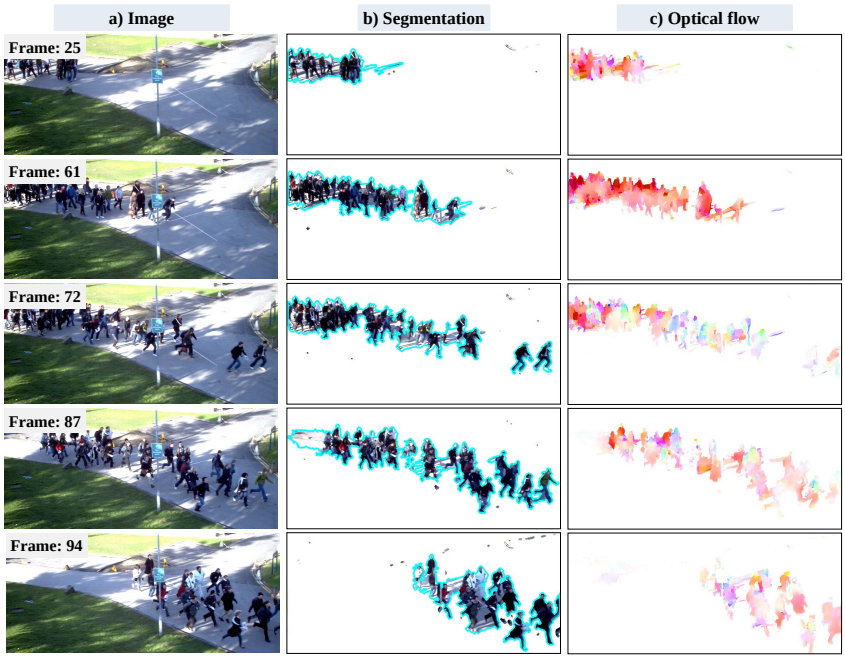


Figure A.12: shows results on PETS2009 [3] dataset. a) input sequence, b) indicates the segmented region, and c) the optical flow computed on segmented region which is presented by using the color encoding scheme [102].

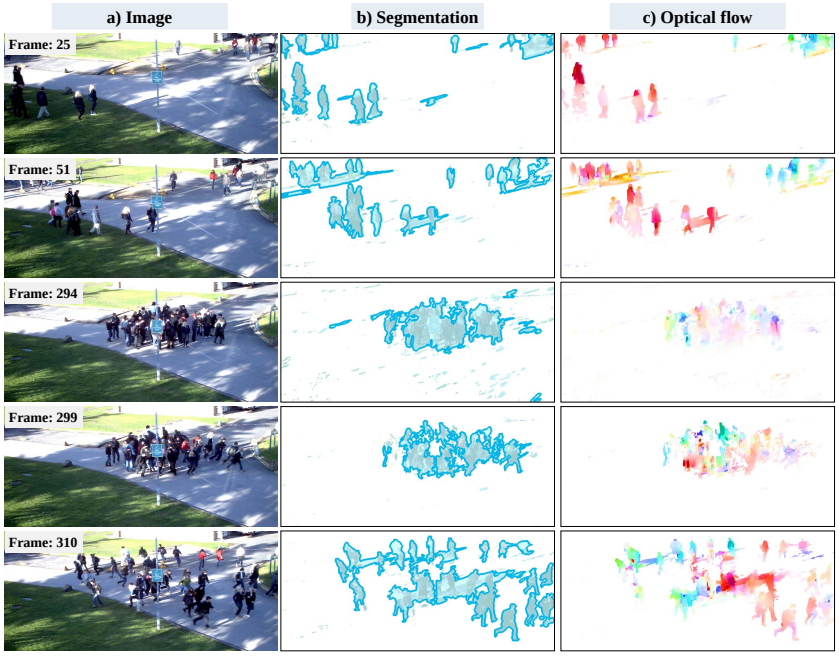


Figure A.13: shows results on PETS2009 [3] dataset. a) input sequence, b) indicates the segmented region, and c) the optical flow computed on segmented region which is presented by using the color encoding scheme [102].

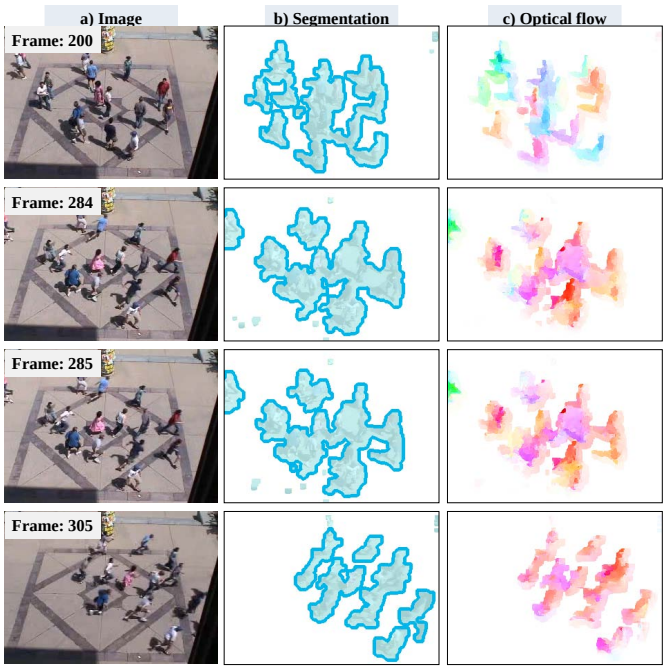


Figure A.14: shows results on UMN [2] dataset. a) input sequence, b) indicates the segmented region, and c) optical flow computed on segmented region which is presented by using color encoding scheme [102].

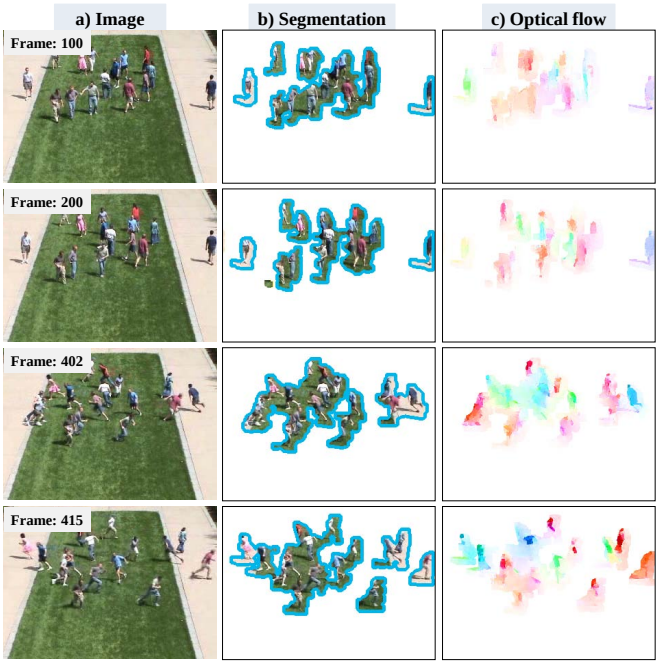


Figure A.15: shows results on UMN [2] dataset. a) input sequence, b) indicates the segmented region, and c) optical flow computed on segmented region which is presented by using color encoding scheme [102].

Bibliography

[1] Ferryman, J.: Performance evaluation of tracking and surveillance (pets) 2006 (2006) *www.cvg.rdg.ac.uk/PETS*2006.

[2] UMN: Detection of unusual crowd activity (2008) *http://mha.cs.umn.edu*.

[3] Ferryman, J., A.Shahrokni: Performance evaluation of tracking and surveillance (pets) 2009 (2009) *www.cvg.rdg.ac.uk/PETS*2009.

[4] Priese, L., Rehrmann, V.: A fast hybrid color segmentation method. In: Proceedings Mustererkennung, DAGM Symposium, Springer Verlag (1993) 297–304

[5] Werlberger, M., Trobin, W., Pock, T., Wedel, A., Cremers, D., Bischof, H.: Anisotropic Huber-L1 optical flow. In: Proceedings of the British Machine Vision Conference (BMVC). (2009) 108.1–108.11

[6] Zhan, B., Monekosso, D., Remagnino, P., Velastin, S., Xu, L.: Crowd analysis: A survey. Machine Vision Application **19** (2008) 345–357

[7] websource: 10 deadliest stampedes in history (2009) *www.bukisa.com/articles/*$28728_1$0$-deadliest-stampedes-in-history$.

[8] Pathan, S.S., Al-Hamadi, A., Michaelis, B.: Crowd behavior detection by statistical modeling of motion patterns. In: International Conference on Soft Computing and Pattern Recognition (win best paper award). (2010) 81–86

[9] Bar-Shalom, Y., Fortmann, T.E.: Tracking and data association. Volume 179 of Mathematics in Science and Engineering. Academic Press Professional, Inc. (1987)

[10] Reid, D.B.: An algorithm for tracking multiple targets. IEEE Transactions on Automatic Control **24** (1979) 843–854

[11] Joachims, T., Finley, T., Yu, C.N.J.: Cutting-plane training of structural svms. Journal of Machine Learning **77** (2009) 27–59

[12] Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. Proc. 18th International Conf. on Machine Learning (2001) 282–289

[13] Cover, T.M., Thomas, J.A.: Elements of information theory. Wiley-Interscience, New York, NY, USA (1991)

[14] Bailey, K.D.: Social entropy theory: An overview. Journal Systemic Practice and Action Research **3** (1990) 365–382

[15] Bouwmans, T., Baf, F.E., Vachon, B.: Background Modeling using Mixture of Gaussians for Foreground Detection - A Survey. Recent Patents on Computer Science **1** (2008) 219–237

[16] Forsyth, D., Ponce, J.: Chapter 15: Segmentation using Clustering Methods. 1 edn. Prentice Hall (2002)

[17] Al-Hamadi, A., Michaelis, B.: An intelligent paradigm for multi-objects tracking in crowded environment. JDIM **4** (2006) 184–191

[18] Cheung, S.C., Kamath, C.: Robust background subtraction with foreground validation for urban traffic video. EURASIP Journal of Application Signal Process. **2005** (2005) 2330–2340

[19] Skarbek, W., Koschan, A.: Colour image segmentation a survey. Technical report, Technical Report, Technical University of Berlin, Department of Computer Science (1994)

[20] Cucchiara, R., Grana, C., Piccardi, M., Prati, A.: Detecting moving objects, ghosts, and shadows in video streams. IEEE Transactions on Pattern Analysis and Machine Intelligence **25** (2003) 1337–1342

[21] Elgammal, A.M., Davis, L.S.: Probabilistic framework for segmenting people under occlusion. Proceedings of IEEE International Conference on Computer Vision **2** (2001) 145

[22] McFarlane, N.J.B., Schofield, C.P.: Segmentation and tracking of piglets in images. Machine Vision and Applications **8** (1995) 187–193

[23] Remagnino, P., Baumberg, A., Grove, T., Hogg, D.C., Tan, T., Worrall, A., Baker, K.: An integrated traffic and pedestrian model-based vision system. In: Proceedings of British Machine Vision Conference. (1997) 380–389

[24] Karmann, K.P., Brandt, A.: Moving object recognition using an adaptive background memory. In: Proceedings of Time-Varying Image Processing and Moving Object Recognition, V. Cappellini, ed. 2, Elsevier Science Publishers B.V. (1990) 289–307

[25] Koller, D., Weber, J., Huang, T., Malik, J., Ogasawara, G., Rao, B., Russell, S.: Towards robust automatic traffic scene analysis in real-time. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. (1994) 126–131

[26] Stauffer, C., Grimson, W.E.L.: Learning patterns of activity using real-time tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 747–757

[27] Wren, C., Azarbayejani, A., Darrell, T., Pentland, A.: Pfinder: real-time tracking of the human body. IEEE Conference on Automatic Face and Gesture Recognition (1996) 51

[28] Mclachlan, G., Peel, D.: Finite Mixture Models. 1 edn. Wiley Series in Probability and Statistics. Wiley-Interscience (2000)

[29] Friedman, N., Russell, S.: Image segmentation in video sequences: A probabilistic approach. In: Conference on Uncertainty in AI. (1997) 175–181

[30] Zivkovic, Z., Heijden, F.: Recursive unsupervised learning of finite mixture models. IEEE Transactions on Pattern Analysis and Machine Intelligence **26** (2004) 651–656

[31] Cheng, J., Yang, J., Zhou, Y., Cui, Y.: Flexible background mixture models for foreground segmentation. Journal of Image and Vision Computing **24** (2006) 473–482

[32] Power, P.W., Schoonees, J.A.: Understanding background mixture models for foreground segmentation. In: Proceedings of International Conference of Image and Vision Computing. (2002) 267

[33] Pathan, S.S., Al-Hamadi, A., Michaelis, B.: Oif - an online inferential framework for multi-object tracking with kalman filter. In: Computer Analysis of Images and Patterns, 13th International Conference (CAIP). (2009) 1087–1095

[34] Boult, X.G., Gao, X., Ramesh, V.: Error analysis of background adaption. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. (2000) 503–510

[35] Grimson, W., Stauffer, C., Romano, R., Lee, L.: Using adaptive tracking to classify and monitor activities in a site. In: Proceedings of IEEE CVPR. (1998) 22–29

[36] Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In: IEEE Conference on CVPR. Volume 2. (1999) 246–252

[37] Ziou, D., Bouguila, N., Allili, M., El-Zaart, A.: Finite gamma mixture modelling using minimum message length inference: application to sar image analysis. International Journal of Remote Sensing **30** (2009) 771–792

[38] Bishop, C.M.: Mixture models and em. In: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc. (2006) 423–454

[39] Comaniciu, D., Ramesh, V., Meer, P.: Real-time tracking of non-rigid objects using mean shift. In: Proceedings IEEE Conference on Computer Vision and Pattern Recognition. (2000) 142–149

[40] Pathan, S.S., Al-Hamadi, A., Michaelis, B.: Integrating statistical and cognitive model for multi-object tracking in realistic scenarios. In: In International Conference of Image and Vision Computing. (2010)

[41] Gonzalez, R.C., Woods, R.E.: Digital Image Processing. 2nd edn. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (2001)

[42] Albiol, A., Silla, M., Albiol, A., Mossi, J.: Video analysis using corner motion statistics. In: Performance Evaluation of Tracking and Surveillance Workshop at CVPR. (2009) 31–37

[43] Horn, B.K., Schunck, B.G.: Determining optical flow. Technical report, Cambridge, MA, USA (1980)

[44] Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proceedings of the International Joint Conference on Artificial intelligence. (1981) 674–679

[45] Jones, D.G., Malik, J.: A computational framework for determining stereo correspondence from a set of linear spatial filters. In: Image and Vision Computing. (1992) 395–410

[46] Weiss, Y., Adelson, E.H.: Perceptually organized em: A framework for motion segmentation that combines information about form and motion. Technical report, Technical Report 315, M.I.T Media Lab (1995)

[47] Fleet, D.J., Jepson, A.D., Jenkin, M.R.M.: Phase-based disparity measurement. International Journal of CVGIP: Image Understanding **53** (1991) 198–210

[48] Liu, C.: Beyond Pixels: Exploring New Representations and Applications for Motion Analysis. Doctoral Thesis, Massachusetts Institute of Technology (2009)

[49] Black, M., P. Anandan, P.: The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. Computer Vision and Image Understanding **63** (1996) 75–104

[50] Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: Proceedings of European Conference on Computer Vision. Volume 4. (2004) 25–36

[51] Alvarez, L., Deriche, R., Papadopoulo, T., Sánchez, J.: Symmetrical dense optical flow estimation with occlusions detection. International Journal of Computer Vision **75** (2007) 371–385

[52] Bruhn, A., Weickert, J., Schnörr, C.: Combining the advantages of local and global optic flow methods. In: Proceedings of DAGM Symposium on Pattern Recognition, Springer-Verlag (2002) 454–462

[53] Sand, P., Teller, S.: Particle video: Long-range motion estimation using point trajectories. International Journal of Computer Vision **80** (2008) 72–91

[54] Scharstein, D., Szeliski, R.: A database and evaluation methodology for optical flow (2007) $http://vision.middlebury.edu/flow/$.

[55] Efros, A.A., Berg, A.C., Mori, G., Malik, J.: Recognizing action at a distance. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society (2003) 726–

[56] Chaudhry, R., Ravichandran, A., Hager, G., Vidal, R.: Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society (2009) 1932–1939

[57] Saleemi, I., Hartung, L., Shah, M.: Scene understanding by statistical modeling of motion patterns. In: CVPR. (2010) 2069–2076

[58] Ali, S., Shah, M.: A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. (2007) 1–6

[59] Mehran, R., Oyama, A., Shah, M.: Abnormal crowd behavior detection using social force model. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (2009) 935–942

[60] Dee, H., Fraile, R., Hogg, D., Cohn, A.: Modelling scenes using the activity within them. In: Spatial Cognition VI. Learning, Reasoning, and Talking about Space. (2008) 394–408

[61] Pathan, S.S., Al-Hamadi, A., Michaelis, B.: Multi-object tracking using semantic analysis and kalman filter. In: In Proceedings of the 6th IEEE International Symposium on Image and Signal Processing and Analysis. (2009) 271–276

[62] Dee, H.M., Velastin, S.A.: How close are we to solving the problem of automated visual surveillance?: A review of real-world surveillance, scientific progress and evaluative mechanisms. Machine Vision Application **19** (2008) 329–343

[63] Cox, I.J., Hingorani, S.L.: An efficient implementation of reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **18** (2002) 138–150

[64] Isard, M., Maccormick, J.: Bramble: a bayesian multiple-blob tracker. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2001) 34–41

[65] Smith, K., Gatica-Perez, D., Odobez, J.M.: Using particles to track varying numbers of interacting people. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society (2005) 962–969

[66] Ryoo, M.S., Aggarwal, J.K.: Observe-and-explain: A new approach for multiple hypotheses tracking of humans and objects. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2008)

[67] Guo, Y., Hsu, S., Sawhney, H.S., Kumar, R., Shan, Y.: Robust object matching for persistent tracking with heterogeneous features. IEEE Transaction Pattern Analysis Machine Intelligence **29** (2007) 824–839

[68] Gheissari, N., Sebastian, T.B., Hartley, R.: Person reidentification using spatiotemporal appearance. In: Proceedings IEEE Conference on Computer Vision and Pattern Recognition - Volume 2, IEEE Computer Society (2006) 1528–1535

[69] Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society (2005) 886–893

[70] Pathan, S.S., Al-Hamadi, A., Krell, G., Michaelis, B.: Resolving data-association uncertainty - in mutli-object tracking through qualitative modules. In: Proceedings of the Fifth International Conference on Computer Vision Theory and Applications. (2010) 461–466

[71] Theodoridis, S., Koutroumbas, K.: Pattern Recognition, Fourth Edition. 4th edn. Academic Press (2008)

[72] Rosenfeld, A.: Patrick henry winston (editor), the psychology of computer vision. Artificial Intelligence **7** (1976) 279–282

[73] Haritaoglu, I., Harwood, D., Davis, L.: W4: A real time system for detecting and tracking people. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (1998) 962

[74] Sherrah, J., Gong, S.: Resolving visual uncertainty and occlusion through probabilistic reasoning. In: In Proceedings of the British Machine Vision Conference. (2000) 252–261

[75] Bennett, B., Magee, D., Cohn, A.G., Hogg, D.: Enhanced tracking and recognition of moving objects by reasoning about spatio-temporal continuity. Image Vision Computer **26** (2008) 67–81

[76] Halpern, J.Y.: An analysis of first-order logics of probability. Artificial Intelligence **46** (1990) 311–350

[77] Helbing, D., Johansson, A., Al-Abideen, H.Z.: The dynamics of crowd disasters: An empirical study. An Empirical Study. Phys. Rev **75** (2007) 046109

[78] Zhao, T., Nevatia, R.: Bayesian human segmentation in crowded situations. In: Proceeding of IEEE Conference on Computer Vision and Pattern Recognition. (2003) 406–413

[79] Tu, P.H., Rittscher, J.: Crowd segmentation through emergent labeling. In: Proceedings of the European Conference on Computer Vision Workshop SMVP. (2004) 187–198

[80] Brostow, G.J., Cipolla, R.: Unsupervised bayesian detection of independent motion in crowds. In: Proceeding of IEEE Conference on Computer Vision and Pattern Recognition. (2006) 594–601

[81] Stalder, S., H.Grabner, , Gool, L.V.: Exploring context to learn scene specific object detectors. In: IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, Miami, USA. (2009) 63–70

[82] Khan, Z.: MCMC-based particle filtering for tracking a variable number of interacting targets. IEEE Transaction of Pattern Analysis and Machine Intelligence **27** (2005) 1805–1918

[83] Betke, M., Hirsh, D.E., Bagchi, A., Hristov, N.I., Makris, N.C., Kunz, T.H.: Tracking large variable numbers of objects in clutter. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society (2007)

[84] Lin, W.C., Liu, Y.: A lattice-based mrf model for dynamic nearregular texture tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **29** (2007) 777–792

[85] Antonini, G., Martinez, S., Santiago, V., Bierlaire, M., Thiran, J.: Behavioral priors for detection and tracking of pedestrians in video sequences. International Journal of Computer Vision **69** (2006) 159–180

[86] Rodriguez, M., Ali, S., Kanade, T.: Tracking in unstructured crowded scenes. In: Proceeding of IEEE International Conference on Computer Vision. (2009) 1389–1396

[87] Wu, S., Moore, B.E., Shah, M.: Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes. In: IEEE Conference on Computer Vision and Pattern Recognition. (2010)

[88] Boghossian, A., Velastin, A.: Motion-based machine vision techniques for the management of large crowds. In: Proceedings IEEE International Conference on ICECS Electronics, Circuits and Systems. Volume 2. (2002) 961–964

[89] Andrade, E.L., Scott, B., Fisher, R.B.: Hidden markov models for optical flow analysis in crowds. In: Proceedings of Conference on Pattern Recognition, IEEE Computer Society (2006) 460–463

[90] Kratz, L., Nishino, K.: Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. (2009)

[91] Helbing, D., Molnar, P.: Social force model for pedestrian dynamics. Physical Review E **51** (1995) 4282

[92] Benabbas, Y., Ihaddadene, N., Djeraba, C.: Global analysis of motion vectors for event detection in crowd scenes. In: Performance Evaluation of Tracking and Surveillance workshop at CVPR. (2009) 109–116

[93] Chan, A.B., Morrow, M., Vasconcelos, N.: Analysis of crowded scenes using holistic properties. In: Performance Evaluation of Tracking and Surveillance workshop at CVPR. (2009) 31–37

[94] Mahadevan, V., Li, W., Bhalodia, V., Vasconcelos, N.: Anomaly detection in crowded scenes. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. (2010) 1975–1981

[95] Hayman, E., Eklundh, J.O.: Probabilistic and voting approaches to cue integration for Figure-Ground segmentation. (2002) 469–486

[96] Alper, Y., Omar, J., Mubarak, S.: Object tracking: A survey. ACM Computing Surveys **38** (2006) 13

[97] Jain, A.K., Duin, R.P.W., Mao, J.: Statistical pattern recognition: A review. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 4–37

[98] Shimazaki, H., Shinomoto, S.: A method for selecting the bin size of a time histogram. Journal of Neural Computing **19** (2007) 1503–1527

[99] Ross, A., Govindarajan, R.: Feature Level Fusion Using Hand and Face Biometrics. In: Proceedings of SPIE. (2005) 196–204

[100] Jepson, A.D., Fleet, D.J., El-Maraghi, T.F.: Robust online appearance models for visual tracking. IEEE Computer Society Conference on Computer Vision and Pattern Recognition **1** (2001) 415

[101] Bishop, C.M.: Introduction. In: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc. (2006) 1–54

[102] Group, M.V.: Color encoding scheme for optical flow (2009) $http://vision.middlebury.edu/flow/submit$.

[103] Kamada, T., Kawai, S.: An algorithm for drawing general undirected graphs. Journal Information Processing Letters **31** (1989) 7–15

[104] Veenman, C.J., Reinders, M.J.T., Backer, E.: Resolving motion correspondence for densely moving points. IEEE Trans. Pattern Anal. Mach. Intell. **23** (2001) 54–72

[105] Yilmaz, A., Li, X., Shah, M.: Contour-based object tracking with occlusion handling in video acquired using mobile cameras. IEEE Transactions on Pattern Analysis and Machine Intelligence **26** (2004) 1531–1536

[106] Zhang, L., Li, Y., Nevatia, R.: Global data association for multi-object tracking using network flows. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. (2008)

[107] Theodoridis, S., Koutroumbas, K.: Pattern Recognition, Fourth Edition. 4th edn. Academic Press (2008)

[108] Russell, S.J., Norvig, P.: Logical Agents. 2 edn. Pearson Education (2003)

[109] Kalman, R.E.: A new approach to linear filtering and prediction problems. Transactions of the ASME–Journal of Basic Engineering **82** (1960) 35–45

[110] Welch, G., Bishop, G.: An introduction to the kalman filter. Technical report, University of North Carolina at Chapel Hill. (1995)

[111] Sun, D., Roth, S., Black, M.J.: Secrets of optical flow estimation and their principles. In: In Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). (2010) 2432–2439

[112] Shannon, C.E. In: A Mathematical Theory of Communication. University of Illinois Press (1949)

[113] Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y.: Large margin methods for structured and interdependent output variables. J. Mach. Learn. Res. **6** (2005) 1453–1484

[114] Shapovalov, R., Velizhev, A.: Cutting-plane training of non-associative markov network for 3d point cloud segmentation. In: Proceedings of International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, IEEE Computer Society (2011) 1–8

[115] Hempstalk, K., Frank, E., Witten, I.H.: One-class classification by combining density and class probability estimation. In: Proceedings of European Conference on Machine Learning and Knowledge Discovery in Databases, Springer-Verlag (2008) 505–519

Curriculum Vitae

Name	Saira Saleem Pathan
Date of Birth	Jan 27, 1980 in Hyderabad
Nationality	Pakistani
Status	Married
Address	W.-Rathenau Strasse 19, 101, 39106 Magdeburg
Email	saira.pathan@ovgu.de
Education	1999 - 2003 Bachelor of Engineering in Computer Systems Engineering , Department of Computer Systems, Mehran UET, Jamshoro Pakistan **(3rd Position)** 2003 - 2005 Master of Engineering in Communication Systems and Networks, Institute of Information Technology, Mehran UET, Jamshoro Pakistan **(GPA: 3.8)** 2008-present PhD Research, IESK, Otto-von-Guericke University Magdeburg, Germany
Professional Experience	Position: Lecturer since December 2003, Department of Computer System and Software Engineering, Mehran UET, Jamshoro Pakistan

Magdeburg, Feb. 22, 2012
Saira Saleem Pathan

Related Publications

The presented thesis has the following international peer-reviewed journals and conference papers:

Journal Publication

1. Saira Saleem Pathan, Omer Rashid, Ayoub Al-Hamadi, and Bernd Michaelis: **"Multi-Object Tracking in Dynamic Scenes by Integrating Statistical and Cognitive Approaches"**, International Journal of Computer Science Issues (accepted). Impact factor: 0.24

2. Saira Saleem Pathan, Ayoub Al-Hamadi, and Bernd Michaelis: **"Crowd Behavior Analysis and Anomaly Detection by Statistical Modeling of Motion Patterns"**, International Journal of Data Mining, Modeling and Management (accepted).

3. Saira Saleem Pathan, Ayoub Al-Hamadi, and Bernd Michaelis: **"Intelligent Feature-guided Multi-Object Tracking in Monocular Color Image Sequences Using Kalman Filter"**, Journal of Computing, Vol 2, issue 11, 2010, pp 6-13. (ISSN 2151-9617).

Conference Publication

1. Saira Saleem Pathan, Ayoub Al-Hamadi, Omer Rashid,Bernd Michaelis: **" Learning a priori threshold to initialize flow-based adaptive mixture model for dynamic scene segmentation"**, IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS), 2011, NJ Piscataway, pp. 691-695.

2. Saira Saleem Pathan, Ayoub Al-Hamadi, and Bernd Michaelis: **"Crowd Behavior Detection by Statistical Modeling of Motion Patterns"**, International Conference on Soft Computing and Pattern Recognition, 2010, Cergy Pontoise/Paris, France, pp 81-86.

3. Saira Saleem Pathan, Ayoub Al-Hamadi and Bernd Michaelis: **"Incorporating Social Entropy for Crowd Behavior Detection Using SVM"**, 6th International Symposium on Visual Computing, Nov 29- Dec1, 2010. Las Vegas, USA, pp. 153-162.

4. Saira Saleem Pathan, Ayoub Al-Hamadi, and Bernd Michaelis: **"Using Conditional Random Field For Crowd Behavior Analysis"**, in International Workshop on Video Event Categorization, Tagging and Retrieval (VECTaR 2010) in conjunction with Asian Conference on Computer Vision (ACCV), Queenstown, New Zealand, pp. 370–379.

5. Saira Saleem Pathan, Ayoub Al-Hamadi, and Bernd Michaelis: **"Integrating Statistical and Cognitive Model for Multi-Object Tracking in Realistic Scenarios"**, In 25th International Conference of Image and Vision Computing New Zealand Queenstown, November 2010.

6. Saira Saleem Pathan, Ayoub Al-Hamadi, Gerald Krell and Bernd Michaelis: **"Resolving Data-Association uncertainty in Mutli-object Tracking through Qualitative Modules"**, VISAPP 2010 - International Conference on Computer Vision Theory and Applications, France, pp. 461-466.

7. Mahmoud Elmezain, Ayoub Al-Hamadi, Saira Saleem Pathan, and Bernd Michaelis: **"Spatio-Temporal Feature Extraction-Based Hand Gesture Recognition for Isolated American Sign Language and Arabic Numbers"**, IEEE International Symposium on Image and Signal Processing and Analysis (ISPA), September 6-18, 2009. Salzburg, Austria, pp. 254-259.

8. Saira Saleem Pathan, Ayoub Al-Hamadi, Mahmoud Elmezain, and Bernd Michaelis: **"Feature-supported Multi-hypothesis Framework for Multi-object Tracking using Kalman Filter"**, International Conference on Computer Graphics, Visualization and Computer Vision, WSCG, Feb. 2-5, 2009. Plzen, CZ, pp.197-202.

9. Saira Saleem Pathan, Ayoub Al-Hamadi, and Bernd Michaelis: **"Intelligent Feature-guided Multi-object Tracking Using Kalman Filter"**, The 2nd IEEE International Conference on Computer, Control & Communication (IEEE-IC4 2009), February 17-18, 2009. – Karachi, pp.1-6.

10. Saira Saleem Pathan, Ayoub Al-Hamadi, Tobias Senst and Bernd Michaelis, **"Multi-Object Tracking Using Semantic Analysis and Kalman Filter"**, In Proceedings of the 6th International Symposium on Image and Signal Processing and Analysis, 2009, pp. 271-276.

11. Saira Saleem Pathan, Ayoub Al-Hamadi, and Bernd Michaelis: **"OIF- An Online Inferential Framework for Multi-object Tracking with Kalman Filter"**, Computer Analysis of Image and Patterns, CIAP 2009; LNCS 5702, Münster, Germany, 2009, pp. 1087-1095.

12. Ayoub Al-Hamadi, Saira Saleem Pathan, Uli Homberg, and Bernd Michaelis: **"Multi-Object Tracking Based on Particle Filter and Data Association in Color Image Sequences"**, International Conference on Computer Vision and Graphics 2008, Nov. 10-12, Warsaw, Poland, pp. 133-142.

13. Ayoub Al-Hamadi, Robert Niese, Saira Saleem Pathan, and Bernd Michaelis: **"Geometric and Optical Flow Based Method for Facial Expression Recognition in Color Image Sequences"**, International Conference on Computer Vision and Graphics 2008, Nov. 10-12,Warsaw, Poland, pp. 228-238.