Università degli Studi di Milano-Bicocca
Dipartimento di Informatica, Sistemistica e Comunicazione
Dottorato di Ricerca in Informatica – XXVIII Cycle

Automatic Detection and Computer Vision Analysis of Flow Dynamics and Social Groups in Pedestrian Crowds

Sultan Daud Khan
Ph.D. Thesis

Thesis supervisor: Prof. Dr. Giuseppe Vizzari, Dr. Saleh Basalamah
Thesis tutor: Prof. Dr. Stefania Bandini

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Crowds of pedestrians can be considered as complex entities from different points of view: the variety individual and collective behaviours that take place in a crowd, the composite mix of competition for the space shared by pedestrians but also the collaboration due to the not necessarily explicit but often shared (at least in a given scenario) social norms, the possibility to detect self-organization and emergent phenomena they are all indicators of the intrinsic complexity of a crowd. The relevance of human behaviour, and especially of the movements of pedestrians, in built environment in normal and extraordinary situations (e.g. evacuation), and its implications for the activities of architects, designers and urban planners are apparent, especially considering dramatic episodes such as terrorist attacks, riots and fires, but also due to the growing issues in facing the organization and management of public events (ceremonies, races, carnivals, concerts,parties/social gatherings, and so on) and in designing naturally crowded places (e.g. stations, arenas, airports). The phenomena of crowd like sports, festivals, concerts, political gatherings etc, are mostly observed in urban areas, which attracts hundreds of thousands people. Pedestrian and crowd modelling research context regards events in which a large number of people may be gathered or bound to move in a limited area; this can lead to serious safety and security issues for the participants and the organisers. The understanding of the dynamics of large groups of people is very important in the design and management of any type of public events. In addition to safety and security concerns, also the comfort of event participants is another aim of the organisers and managers of crowd related events. Large people gatherings in public spaces (like pop-rock concerts or religious rites participation) represent scenarios in which crowd dynamics can be quite complex due to different factors (the large number and heterogeneity of participants, their interactions, their relationship with the performing artists and also exogenous factors like dangerous situations and any kind of different stimuli present in the environment. Such crowding phenomena poses serious challenges to public safety and crowd management. Therefore analysis of crowd is crucial for solving real world problems. Researchers from different communities like sociology, civil, physics and computer science are studying crowding phenomena from different angles. Besides these efforts, computer vision research community developing algorithms that can automatically understand the crowd dynamics in the real-world scenes. Despite these efforts, computer vision research community have not achieved the desired level of applicability and robustness. This is due to the fact that the algorithms are based on particular assumptions which are often violated in real-world environment.

From the computer vision's point of view, there are three traditional preprocessing step for performing crowd analysis automatically. i) object detection, ii) tracking iii) behaviour understand based on the analysis of extracted trajectories. In real world, a crowd is more than sum of the few individuals; the task for computer vision become more complex when the number of individuals in the scene increases. This can be observed from the fact that human response to high density crowd image is much slower than to a non-crowded image. For example, a human can easily detect, count and track few individuals in the scene but when presented with an image containing hundreds of thousands people, will need large amount of time to count. This highlights the fact that a simple extension of computer vision's algorithms that are designed to detect and track few individuals in the scene can be be applicable to complex scene. Therefore, for analysis crowded scenes, holistic approaches mostly based on the optical flow,e.g, finding motion flow patterns, segmentation of crowd flows are adopted.

In this thesis, i select some of the challenging problems regarding understanding crowd dynamics and i develop some methods to address those problems. The methods propose in this thesis out perform the state-of-the-art methods and i rigorously evaluated the propose methods by considering complex and challenging scenarios of the real world scenes.

1.2 Motivation

As the population of world is increasing and ever more located in urban areas, public safety is becoming a problem in most crowded areas of the big cities. Mass events like those related to sports, festivals, concerts, and carnivals attract thousands of people in constrained environments, therefore adequate safety measures must be adopted. Some of the examples of mass events are illustrated in Figure. Despite all safety measures, crowd disasters still occur frequently. Summary of different incidents of crowd disaster can be found in Table 1.1

The reason of these disasters is different and conflicting motion patterns that influence the crowd. One of the eye witness of recent Mina's incident reported in a newspaper, *"Because those returning in opposite direction of the surging crowd, there was a stampede"* A crowd is composed of small groups of people and these small groups arise due to interdependence among its members. This interdependence among the members may be as result of social relationship or motivated by a common goals. The examples of group that arise due to

Figure 1.1: Examples of crowd

Table 1.1: Crowd disaster

Year	Place	Deaths
2015	Mina, Saudi Arabia	> 750
2011	Stadium, Bamako(Mali)	> 36
2011	Pilgrimage, Kerala(India)	102
2010	Loveparade, Germany	21
2010	Water festival, Combodia	> 375
2006	Stadium, Yemen	> 51
2005	Religious procession, Iraq	> 640
1990	Pilgrimage, Saudi Arabia	1426
1982	Stadium, Russia	340

social relationship are member of family or friends; these groups can be called long term coherent groups because they maintain the structure for long period of time. There are, however, other types of groups essentially motivated by a common goal, like reaching a certain point of the environment; these groups can be called short term coherent groups because they discontinue their cohesion after completing the goals (e.g. reaching an exit, completing a movement). Detecting the second kind of group, essentially associated to a certain flow of pedestrians in the environment, can be important to be able to prevent conflict situations.

Due to the complex dynamics of the crowd, crowd management is becoming a daunting job where huge effort from the security staff is required to manage the potentially problematic situations. For example, During Hajj, every year government of saudia arabia deployed more than 100,000 security personnel. In high density crowded areas, surveillance cameras are generally installed in different locations that can even cover the whole crowd scene. Detecting specific activities in real-time videos is the task of analysts sitting in surveillance room and watching over multiple Tv screens. Such manual analysis of high density crowds is a tedious job and usually prone to errors. For instance, more than 5,000 surveillance cameras are mounted on different locations in Mina. Still it could not help in preventing the disaster. Therefore we need automatic analysis of the crowd which can reliably estimate the density of the crowd and detect specific activities. Creating such kind of virtual analyst has become the focus of many researchers. This research has a wide range of application domain in crowd management, public space design, underwater fishes analysis (and animal behaviour studies in general), and cell population analysis.

In video surveillance, "detection and tracking" of pedestrians are the core technologies. The main concern of the state-of-the-art methods is to localise the moving objects, track the objects for some duration and understand the scene semantics. Understanding the scene semantics together with tracking help in detecting abnormal behaviours in the scene. The application of these methods to high density crowded scenes is limited because when the crowd density increases, these methods likely to fail detecting and tracking the moving objects and hence unable to understand the behaviour of the crowd. Limited research has been reported in literature that can provide good models for high density crowded scenes that ultimately provide useful information for crowd management. One of the reason for the lack of interest and efforts in this direction is the complexity and challenges inherent in high density crowded situations. I discuss some of these complexities and challenges in the following section

1.3 Challenges

Most important challenges that need to be addressed for understanding the dynamics of high density crowds are:

1. In high density crowded scenes, detection of individual becomes very hard due to less number of pixels/person. The number of pixels/person have an indirect relationship with the density of crowd. The larger the crowd density, the few will be the pixels/person, that makes the tasking of detecting and tracking people in such situations very hard. The information about the appearance of individual further disturbed due to the constant interactions among the individuals. Therefore, instead of detecting and tracking individuals in such situations, researchers adopted holistic approaches to understand the crowd dynamics.

2. Physical characteristics of a scene can become the source of occlusion resulting in the loss of information about the considered object. Moreover, complex interactions among the individuals in the scene cause both temporal and spatial occlusions.

3. The individuals in the crowds exhibits different behaviours and usually goal directed. This makes very challenging to figure out an appropriate level of granularity to model crowd dynamics.

4. Another challenge in crowded scene analysis is to detect specific crowd behaviours. One can learn these behaviours but it may be possible that in other surveillance cameras these behaviours have limited instances and we have to learn more behaviours.

1.4 Contributions

In this thesis, I developed algorithms for (i) *Crowd Flow Segmentation and Crowd Counting*, (ii) *Crowd Behaviour understanding* , and (iii) *Social Group Detection in Crowd*. In the first two approaches, we considered high density scenarios with more than hundreds of thousands people per frame, while our third algorithm is applicable to low density situations, where the crowd density is 50-60 people per frame. I briefly introduce each of our proposed method and our contributions in the following.

Crowd Flow Segmentation and Crowd Counting

The first algorithm proposed in this thesis automatically segment the crowd in different segments based on their orientations and estimate the number of people in each flow segment. I carried out this analysis by considering very high density crowds where it is very hard to detect and track the individuals. Unlike the traditional methods of video surveillance, I employed holistic approach that captures both the dynamics and structure of the scene. Such holistic approach eliminates the need of localizing the individuals. The proposed approach is applicable in many different situations and it is independent of local conditions and camera viewpoints. Moreover, the proposed method does not require detection and identification of individuals, hence preserving the privacy of the people. The proposed framework consists of four processing blocks, Foreground extraction, segmentation, counting and blob size optimization. The approach starts with generating two foreground masks, one by computing the dense optical flow between two consecutive images, $f_{hs(x,y,t)}$ and one by Gaussian background subtraction, $f_{g(x,y,t)}$. Both these foreground masks serve different purpose. Since the optical flow vector of each pixel has the magnitude and direction values, therefore, we use orientation information of optical flow vectors for crowd flow segmentation by clustering all optical flow vectors by employing K-means clustering algorithm. For the counting framework, we use $f_{g(x,y,t)}$. After generating foreground masks, the next step is the computation of motion field followed by motion field segmentation. After motion field segmentation some small blobs appear representing the small clusters at the boundaries of two opposite flows. We propose blob absorption method in order to get rid of these small clusters. After segmenting the crowd in different segments, we estimate the number of people in each cluster (flow segment) by employing our proposed blob analysis and blob size optimization methods .

Crowd Behaviour Understanding

The second framework developed in this thesis perform crowd behaviour analysis and understanding. In this framework, I developed two novel algorithms, the first able to generate long, dense, reliable and accurate pedestrian trajectories and the second clustering them to generate long term reliable and abstract information describing flows in the whole video. The final results provide directly information characterizing flows but they also represent a starting point for further high-level analyses of crowd behaviour. The approach starts by dividing the input video into multiple *segments* of equal length and duration, considering videos with a constant frame rate. The initial frame of each segment is overlaid by a grid of particles initializing a dynamical system defined by optical flow. Time integration of the dynamical system over a segment of the video provides particle trajectories (*tracklets*) that represent motion patterns in the scene for a certain time interval associated to the analyzed segment. We detect sources, sinks and main flows in the segment (for sake of brevity sometimes we will refer to this information as segment *local track*) by analyzing motion patterns followed by clusters of tracklets, obtained using an unsupervised hierarchical clustering algorithm, where the similarity is measured by the Longest Common Sub-sequence (LCS) metric. Results achieved so far are intrinsically related to a single segment of the analyzed video, associated to a relatively short time frame. To achieve final *global tracks*, covering all the video, we cluster the achieved local tracks through the same hierarchical clustering algorithm. Our main contributions are:

1. generating dense and long trajectories,
2. identifying sources and sinks,
3. understanding behavior of the crowd in the scene by considering full length video,
4. achieve the above results without requiring object detection, tracking, nor training, targeting employment in naturalistic conditions.

Social Groups Detection in Crowds

In the above two proposed frameworks, I focus on the overall crowd by considering a large set of pedestrians without taking into account the importance of social interactions among pedestrians. In many situations, pedestrians do not really walk alone, and researchers observed in most situations pedestrians actually walk in groups. Some interesting forms of social interaction and adaptive behaviours can be observed at the group level and they are growingly investigated in the area of pedestrian and crowd modelling and simulation. Therefore, keeping in view the growing importance of this problem, I proposed an algorithm in this thesis that can automatically detect the social groups in crowds. The approach presented in this thesis starts by extracting trajectory information from the whole video and building an *Association Matrix* that captures the joint probability distribution of start and stop locations of all pedestrians to all other pedestrians in the scene and it adopts a bottom-up hierarchical clustering approach to discover social groups. The main contributions of the work are:

1. instead of considering whole trajectories, we consider only two points (start and stop) making the overall group detection process computationally less expensive and more suitable for real-time operation,

2. our approach does not require training,

3. the usage of *Association Matrix* for discovering couples and *Adjacency Matrix* for discovering groups,

4. our approach requires only one parameter setting..

1.5 Organization of Thesis

The thesis are organized as follows: **Chapter 2** discusses the demography of Hajj. **Chapter 3** discusses the integrated approach of crowd analysis and crowd synthesis for understanding crowd dynamics and the efforts made in this direction. This chapter also discusses classification of crowd studies with particular attention to crowd analysis. Moreover, this chapter discuss different state-of-the-art methods focusing on dealing with similar aspects of crowd as proposed by this thesis. **Chapter 4** presents the crowd flow segmentation and crowd counting framework and intermediate steps involved. **Chapter 5** introduces novel algorithms for extracting correct and reliable point trajectories, that helps in identifying semantic regions in the scene and provide a useful input for crowd behaviour understanding. **Chapter 6** highlights the importance of social groups in understanding overall crowd dynamics and presents novel algorithm for detecting social groups in crowds. The thesis is concluded in **Chapter 7** with summery of contributions and description of future work.

1.6 List of Publications

Part of the thesis has been published in the following list of publications

1. S. D. Khan, G. Vizzari, S. Bandini, S. Basalamah, Detecting dominant motion flows and people counting in high density crowds.

2. S. D. Khan, G. Vizzari, S. Bandini, Identifying sources and sinks and detecting dominant motion patterns in crowds, Transportation Research Procedia 2 (2014) 195–200.

3. M. Arif, S. Daud, S. Basalamah, People counting in extremely dense crowd using blob size optimization, Life Science Journal 9 (3) (2012) 1663–1673.

4. S. D. Khan, G. Vizzari, S. Bandini, S. Basalamah, Detection of social groups in pedestrian crowds using computer vision, in: Advanced Concepts for Intelligent Vision Systems, Springer, 2015, pp. 249–260.

5. S. D. Khan, F. Porta, G. Vizzari, S. Bandini, Estimating speeds of pedestrians in real-world using computer vision, in: Cellular Automata, Springer, 2014, pp. 526–535.

6. S. D. Khan, L. Crociani, G. Vizzari, Integrated analysis and synthesis of pedestrian dynamics: First results in a real world case study.

7. S. D. Khan, Estimating speeds and directions of pedestrians in real-time videos: A solution to road-safety problem.

8. S. D. Khan, L. Crociani, G. Vizzari, Pedestrian and Crowd Studies: Towards the Integration of Automated Analysis and Synthesis, SCS MS Magazine. Vol 4, Issue 3, 2014

9. S. D. Khan, S. Bandini, S. Basalamah, G. Vizzari Analyzing Crowd Behavior in Naturalistic Conditions: Identifying Sources and Sinks and Characterizing Main Flows, Under revision ($2^{nd} round$) Neurocomputing, 2015.

10. S. D. Khan, G. Vizzari, S. Bandini, Facing Needs and Requirements of Crowd Modeling: Towards a Dedicated Computer Vision Toolset, in: Tranportation Granular Flow, 2015.

Chapter 2

Demography of Hajj

2.1 Introduction to Hajj

Hajj is fifth pillar of Islam, where more than two million muslims from various ethnic groups of more than 180 countries worldwided perform a series of ritual activities every year. This is the religious duty for the adult muslims that are physically and financially capable of performing hajj rituals. It is a journey whose goal is to connect the pilgrim with the sacred:it is a moment of transformation from the continuity of the ordinary life (structured) of the persons to a non-structure rituals, that take the risks and the efforts, both spiritual and material, of the pilgrimage.

As any other case of pilgrimage, the Hajj has a dimension of communitas in the sense of Turner communitas that prevails on 'societas' where on the contrary the ordinary, everyday dimension dominates. Hajj pilgrims form an unstructured community where all members are equal. Following Van Gennep's passage model, Turner identified a process of ritual organized in phases determining the transition of an individual from one state to another. Turner noted that in the context of the states of the ritual subjects are often secluded from everyday life and have to spend some time in an inter-structural, liminal situation. In these anti-structure phases there is a preponderance of behavioural attitudes tending to a detachment to some social barriers and norms. In fact, the week before the the beginning of the Hajj, international and regional pilgrims usually in organized groups, start arriving at Makkah. The pilgrims have precise stations and places called Miqat were they 'enter the state of Ihram', which is the sacred state they must enter in order to perform the pilgrimage.

Miqat separates the external and profane space from the internal and sacred area of the Hajj. The first step of every pilgrim is to go through this "passage" by performing the cleansing rituals and wearing the prescribed clothing. Male pilgrims must wear a white seamless garment made up of two pieces of cloth or towels, one covering the body from waist to ankle and the other is thrown over the shoulder; their head must also be uncovered. Women are not prescribed to have a unusual dressing and thus they generally wear a simple dress and a head covering. The pilgrims have to abstain from quarrelling, committing violence to humans or animals, performing specific body care activities (e.g. shaving, cutting nails) and having conjugal relations. The state of Ihram puts the pilgrim in a condition of suspension between the profane world and the sacred one, which will be definitively approached only after having passed through a series of ceremonies, after which the pilgrim will be purified. The Hajj is an act of faith but also an act of peace. In fact the prohibitions tied to the rituals are all related to the promotion of peace. Peace with God, with ones soul, peace with one another and with every other creature. Everyone performing the pilgrimage dresses in the same simple way, observes the same regulations, utters the same supplications at the same time in the same way, for the same end. Once the pilgrims reached the Holy Mosque, they must perform Tawaf and Sa'ay. During Tawaf, the pilgrims walk seven times around the Kaaba in counterclock wise direction. After finishing Tawaf, the pilgrims must walk seven times between hills of Safa and Marwah, located near Kaaba.

2.1.1 Precise days of Hajj

The Hajj takes place on five specified days each year between the 8th and the 12th day (optionally the 13th) of the twelfth month of the islamic calendar, known as Thul-Hijjah. Because of the difference between the islamic calendare which is a lunar calendar and the gregorian calender, used in the western world, the gregorian dates of the Hajj change from year to year. The pilgrimage comprises a precise sequence of rituals conducted at various Holy Sites: Makkah, Mina, Arafat, Muzdalifa and Jamarat.

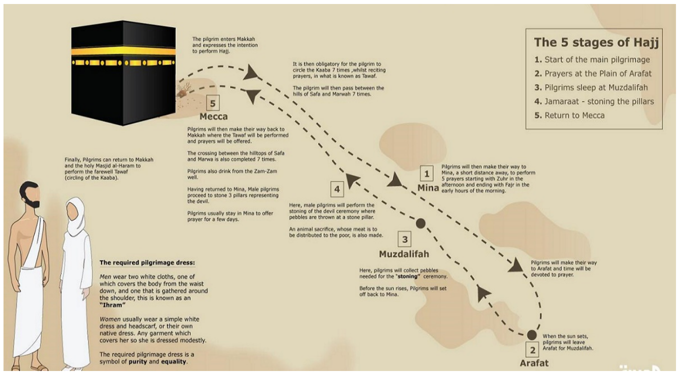
2.1.2 Rituals of Hajj

After 'entering the state of Ihram' the pilgrims perform the Tawaf and the Sa'ay in Masjid al-Haram, the Sacred Mosque, in Makkah, the pilgrims walk seven times around the Kaaba in counterclock wise direction. After finishing Tawaf, the pilgrims must walk seven times between hills of Safa and Marwah, located near Kaaba.

The first day of the Hajj, after Tawaf and Sa'y, the pilgrims go to Mina. They perform five prayers, starting with the noon prayer (Zuhr) and ending with dawn prayer (Fajr) and they collect some of the seventy small pebbles they will need for the "stoning" ceremony.

On the second day of the Hajj pilgrims leave Mina after Dawn Prayer, moving towards the plain of Arafat, where they spend the whole night. The permanency here is also called the Wuquf (i.e. "being and standing" – implicitly meaning before God), which is the central rite of the Hajj. Pilgrims can stay everywhere they want in the plain area of more than 1000 hectares in which Arafat is situated, which also includes the Mountain of Mercy, where it is believed the Prophet Muhammad delivered his Farewell Sermon. When the sun has set pilgrims leave Arafat for Muzdalifah. The pilgrims will stay here from the after sunset of the second day till after the down prayer on the third day. While they stay in Muzdalifah, they have to make sure they have all the pebbles they need for the stoning ceremony and if not they have to collect more. At dawn they offer Fajr (the dawn prayer) and then, before the sun has risen, they set off for Mina.

The third day pilgrims are in Mina for the stoning ceremony called Ramy in which they stone seven times the pillar that represent the devil. The stoning is followed by the animal sacrifice; some pilgrims also cook and eat the killed animal. After this step, they leave the state of Ihram, by shaving and cutting of hair and change into normal clothes. The pilgrims now proceed to the al Masjid al Haram in Makkah to perform Tawaf al-Ifadha. They again circumambulate the Kaaba seven times and then The third day pilgrims are in Mina for the stoning ceremony called Ramy in which they stone seven times the pillar that represent the devil. The stoning is followed by the animal sacrifice; some pilgrims also cook and eat the killed animal. After this step, they leave the state of Ihram, by shaving and cutting of hair and change into normal clothes. The pilgrims now proceed to the al Masjid al Haram in Makkah to perform Tawaf al-Ifadha. They again circumambulate the Kaaba seven times and then offer prayers. After performing again the Sa'ay the pilgrims return to Mina where they spend the night.



The pilgrim enters Makkah and expresses the intention to perform Hajj.

It is then obligatory for the pilgrim to circle the Kaaba 7 times ,whilst reciting prayers, in what is known as Tawaf.

The pilgrim will then pass between the hills of Safa and Marwah 7 times.

The 5 stages of Hajj

1. Start of the main pilgrimage
2. Prayers at the Plain of Arafat
3. Pilgrims sleep at Muzdalifah
4. Jamaraat - stoning the pillars
5. Return to Mecca

Pilgrims will then make their way back to Makkah where the Tawaf will be performed and prayers will be offered.

The crossing between the hilltops of Safa and Marwa is also completed 7 times.

Pilgrims also drink from the Zam-Zam well.

Having returned to Mina, Male pilgrims proceed to stone 3 pillars representing the devil.

Pilgrims usually stay in Mina to offer prayer for a few days.

Finally, Pilgrims can return to Makkah and the holy Masjid al-Haram to perform the farewell Tawaf (circling of the Kaaba).

Pilgrims will then make their way to Mina, a short distance away, to perform 5 prayers starting with Zuhr in the afternoon and ending with Fajr in the early hours of the morning.

Here, male pilgrims will perform the stoning of the devil ceremony where pebbles are thrown at a stone pillar.

An animal sacrifice, whose meat is to be distributed to the poor, is also made.

Here, pilgrims will collect pebbles needed for the **"stoning"** ceremony.

Before the sun rises, Pilgrims will set off back to Mina.

Pilgrims will make their way to Arafat and time will be devoted to prayer.

When the sun sets, pilgrims will leave Arafat for Muzdalifah.

The required pilgrimage dress:

Men wear two white cloths, one of which covers the body from the waist down, and one that is gathered around the shoulder, this is known as an **"Ihram"**

Women usually wear a simple white dress and headscarf, or their own native dress. Any garment which covers her so she is dressed modestly.

The required pilgrimage dress is a symbol of **purity** and **equality**.

Figure 2.1: Rituals of Hajj

On the afternoon of the fourth day and again the following day the pilgrims must again throw seven pebbles at each of the three Jamarat in Mina. Pilgrims can decide to return to al Masjid al Haram to perform the farewell Tawaf after Ramy on the fifth day (12th day of Thul-Hijjah), or stay till the sixth day, performing Ramy for the third time, before returning to al Masjid al Haram for the farewell Tawaf. Farewell Tawaf marks the end of the Hajj. The pilgrims spend whatever time they can within the precincts of al Masjid al Haram and they make the prayers and acts of devotion as they wish.

2.2 Demography of Hajj

Demography is the statistical study of human population. Usually this study involves the measurement of fertility, mortality and migration. Once the data about the these so-called demographic components are gathered, then different sophisticated statistical tools can be used to predict these components of the population. Since we discuss talk about the population of hajj where some components of the demography like fertility rate, etc are not important and we will discuss other important related components like, population size, population diversity, and to some extent, mortality rate in the context of different incidents during Hajj.

Muslims representing 23% of the world's population, and according to report[1], the percentage of muslim population will increase to 29.7%. Figure 2.2 shows that muslims are the only major religious group expected to increase faster. With the possible increase in the muslim population, there will be increase in the population of people performing hajj. The growth of pilgrims population during the past years is illustrated in Figure 2.3. The growth of pilgrims population reported in Figure seems to be inconsistent and could not clearly reflect yearly growing population of the pilgrims. For example, the population of pilgrims in the last three years is lower than previous years. There are couple of reasons that may explain the decrease in the growth, i) The population data in the Figure 2.3 is based on the official data provided by the government. Government has the record of the pilgrims who made the registration for the hajj, and for each country they have establish a quota system, that limits the participation of pilgrims from different countries. Ideally, the quota system is based on the percentage of muslim population of a country. The more muslims population in a country, the higher will be the quota. But the current quota system to some extent is a political. The muslims population in Iran is higher than the quota assigned. The assigned limited quota to Iran, due to possible threat to kingdom and Makkah itself. In 1987, a clash between the pilgrims from Iran and Saudi Arabia security forces occurred, which lead to the deaths of over 400 people. ii)There are many muslims inside and outside the kingdom who travelled secretly through different routes to reach Makkah in order to perform hajj. These out numbered people are unregistered and the government have no record about their count. Therefore, the counts reported in Figure 2.3 are under estimated ignoring the count of unregistered pilgrims. In order to prevent the the flow of un-registered pilgrims, the security personnel deployed at different entry locations of makkah. iii) Saudi Arabian government has started an expansion project 2020 and due to the on-going construction, the government decrease the quota for each country.

[1] The Future of World Religions: Population Growth Projections, 2010-2050

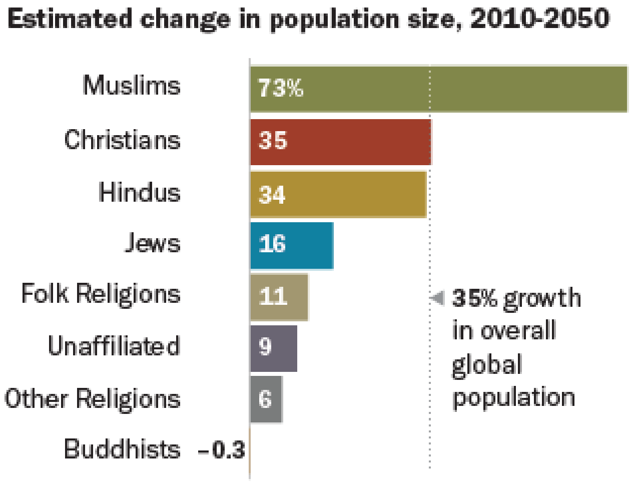


Figure 2.2: Growth rate of different religious groups

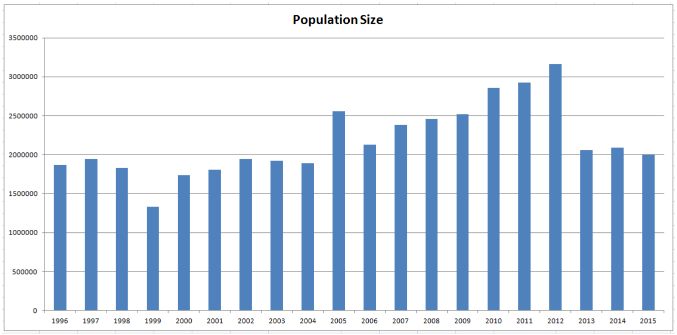


Figure 2.3: Number of pilgrims per Year

2.2.1 Hajj Mortality Rate

The number of pilgrims are increasing every year and since hajj involves unique migration of large number of people moving from one place to another in extreme hot weather within a constraint environment. Such huge migration of people from one place to another while performing rituals often leads to accidents.i.e, stampedes and failures of crowd control. In most of the cases stampede occurs due to movement of conflicting flows (moving in opposite directions), for example, the group of people after finishing stoning the devil ritual return and come in conflict with the group of people going to perform the same ritual. Hence panic spreads among the pilgrims in order to avoid being trampled, and many pilgrims died as a result. The number of pilgrims per year died during stampedes is shown in Figure 2.6. Beside stampedes the rate of natural deaths

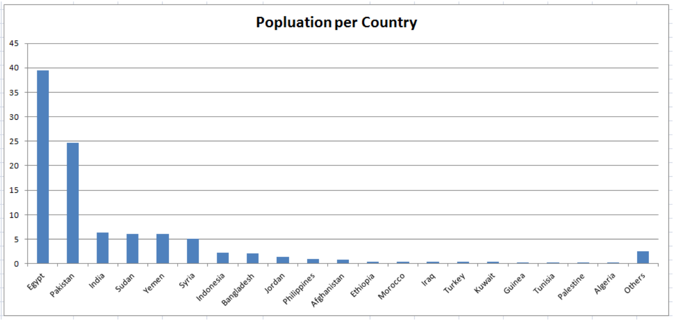


Figure 2.4: Distribution of hajj population per country

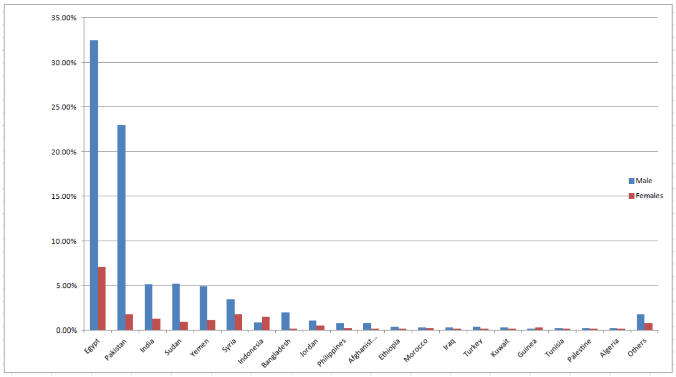


Figure 2.5: Distribution of hajj population per gender

among pilgrims is high, since most of the pilgrims are from developing countries are aged and with poor health. Most deaths are due to the cardiovascular and respiratory diseases. Over the past few years, cardiovascular disease become an significant cause of deaths of most of pilgrims. For example, more than 60% of the Intensive care units (ICUs) of hospitals in Mina, Arafat came from cardiovascular reasons. The percentage of pilgrims admitting to hospitals during hajj specific days is higher as illustrated in Figure 2.7. The percentage of cardiovascular diseases was very high during the hajj 2002, 13.8% admitted to hospitals due to respiratory problems as shown in Table 2.1.

Analysis of the age distribution revealed that admission to hospitals is often dominated by the pilgrims older than 40 years[2]. This is obvious from the fact that cardiovascular diseases are more common in old people and since most of pilgrims are aged people more prone to these diseases.

[2] Khan N.A., Ishag A.M., Ahmad M.S., El-Sayed F.M., Bachal Z.A., Abbas T.G. Pattern of medical diseases and determinants of prognosis of hospitalization during 2005 Muslim pilgrimage Hajj in a tertiary care hospital. A prospective cohort study, Saudi Medical Journal. 2006;27(9):1373-1380

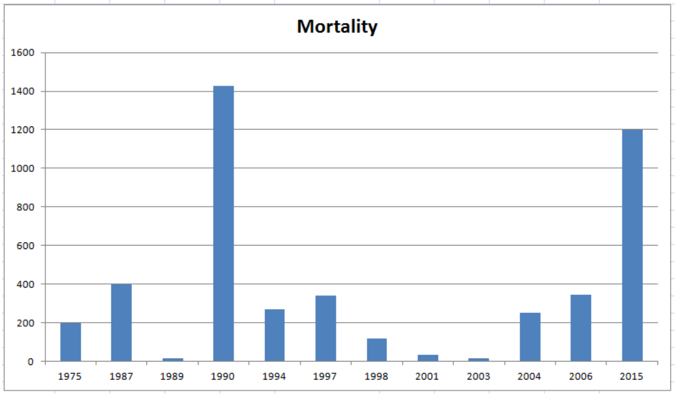


Figure 2.6: Mortality Rate because of stempeds

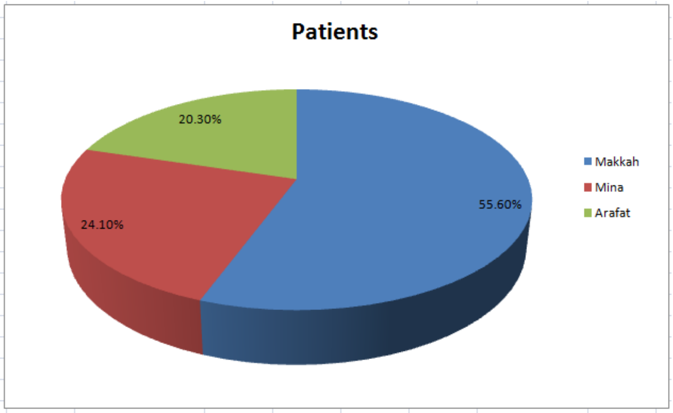


Figure 2.7: Patients admitted to hospitals

2.2.2 Ageing and the hajj

The presence of over two million pilgrims naturally implies the presence of large number of elderly persons that are however carrying out this religious duty with great enthusiasm. Ageing of population is currently one of the most relevant demographic component in industrialized nations, where it is going to produce significant modifications from the economic, social and cultural perspective. This phenomena should not be considered as the cause of negative consequences, but invested to highlight relationships, needs and potentialities that an ageing society is able to express. In particular, it is necessary to reflect on how the social inclusion of elderly people will be guaranteed in future and how to improve their mobility. Mobility is essential for general independence as well as ensuing good health and quality of life, and one of the most relevant and important activities of daily living for maintaining independence. Although Saudia Arabia is not

Table 2.1: Significant causes of death

Diseases	**Mortality**
Cardiovascular	45.8%
Respiratory	13.8%
Traffic accidents	6.4%
Cerebrovascular	3.4%

Table 2.2: Pilgrims age vs Mortality

Age(years)	**Mortality**
Less than 20	0.0%
20-39	3.5%
40-59	2.02%
60-79	67.5%
Greater than 80	8.8%

facing the problems of ageing society but every year they have huge gathering of aged people during hajj. There are couple of reasons of aged people coming for hajj. The first reason is that, most of the pilgrims came for performing hajj from developing countries like egypt, pakistan, india, etc as evident from Figure 2.5. The per capita income of these countries is very low and most of population is living below poverty level. The population of these countries is very high and usually there is only one bread winner, supporting 5 to 6 members of the family. Under these circumstances, people are not financially stable enough to go for the hajj at the early stages of their lives. They usually safe the money for whole of their lives, so at the end, they could go for the hajj. The second reason is that most of them think, although not true from religious point of view, that if they die during performing hajj, which normally happened due the health problems related to ageing discussed above, they would go the heavens.

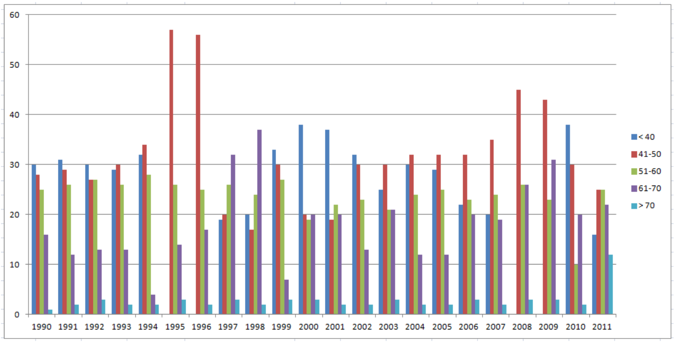


Figure 2.8: Distribution of age

2.3 Transportation in Hajj

Public transportation is the movement of people from one place to another. Usually transportation is base on fixed infrastructure like roads, waterways, railways, station etc. Means of transport is a term used to distinguish different ways to perform transport. In today modern world, the most dominant means of transportation are aviation (by air), ship transport, and land transport. Other traditional means of transportation include, pipelines, cable transport, human and animal powered transport. The modern means of transportation like air, water and land are fastest ways that covers thousands for miles distance in less time. Each means of transportation has fundamentally different infrastructure and require a separate environment. Each means of transportation has separate subsystems. All means of transportation have six subsystems: propulsion, suspension, control, guidance, structural, and support.

Transportation is considered an important subject during hajj, where more than two million people move from place to place performing different rituals. We categorise the transportation in hajj into two; (i) *Global transportation* (ii) *Local transportation.* Pilgrims from different parts of the world, in order to reach Holy City adopt different means of transportation, we call this as a global transportation. Once the pilgrims arrive in the Holy City, they adopted different means of transportation like, buses, vehicles, bikes, trains etc to move in the city and also during hajj. A brief summary of this categorization is illustrated in Figure 2.9.

2.3.1 Global Transportation

Air

Transportation by air is the modern and second fastest means, after space travel. Commercial aeroplanes reach speeds of up to 955 kilometre per hour and longer distances are easily covered in one or a few days. Air transportation is the fastest mean among other public means of transportation but it costly and consumes energy more than others. Moreover, aviation effect the climate and particularly global climate 2-4 times more than other means of transportation. During hajj period, most of the pilgrims reach to the Holy city by using air transportation

Land

Land transportation is the movement of people from one location to another on land usually by road. A road is identifiable path between two or more places. Roads are typically smoothed, paved in order to allow easy travel. The most common road vehicles are auto mobiles, buses, motorcycles and pedestrians. Auto mobiles offer high flexibility and with low capacity, but with high energy use. This means of transportation are the main source of noise and air pollution in urban areas. During hajj period, the pilgrims from middle east countries who share borders with Saudi arabia adopt this mean of transportation. Since this means of transportation is cheaper than aviation.

Water

Ship transport has been the largest means of transport since last many centuries. Due to the availability of modern means of transportation like aviation, the importance of sea travel for passengers has decreased. Transportation by water is cheaper than air. During the early part of previous century, pilgrims used to reach the Holy city by ships sailing through the Red sea. But the popularity of aviation reduce the use of water transportation not only for hajj but in general. Following table shows the percentage of people using different means of transportation to reach Makkah during hajj. As evident from the Table 2.3, almost all pilgrims uses aviation while very small portion of pilgrim uses other means.

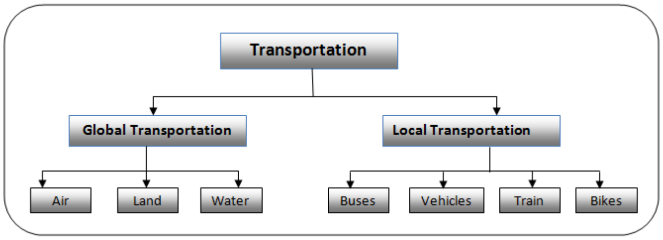


Figure 2.9: Categorization of Transportation in Hajj

Table 2.3: Means of Transportation

Videos	Male	Female	Total
Air	51.60%	43.12%	94.73%
Land	2.44%	1.81%	4.26%
Sea	0.51%	0.49%	1.00%

2.3.2 Local Transportation

Once the pilgrims arrived in the Holy City, they use different means of transportation for different purposes. Usually, most of the pilgrims stay in hotels, and the government and administration of hotels provide buses that take the pilgrims from their residence to the Holy mosque and vice versa. They adopt other means like vehicles to visit around different places (Holy places) of the city. During the hajj days, the common means of transportation are buses, vehicles. But due to extreme crowded situations, these means of transportation get jammed as depicted in Figure and pilgrims either adopt to go by themselves or hire bikes, so that they can easily take them to their destination in order to avoid jam. In order to avoid jam of massive proportions, the government has constructed Mashaer rail line.

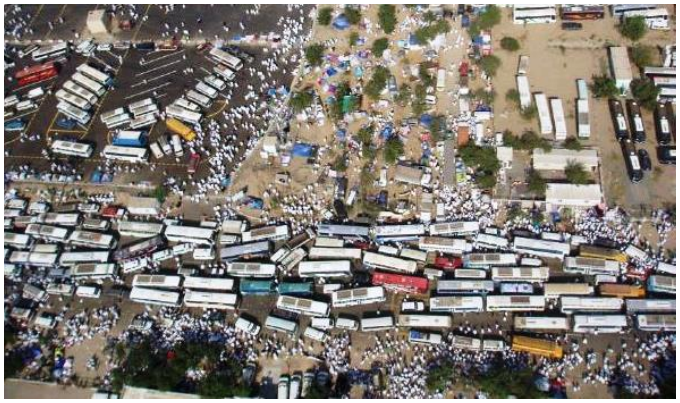


Figure 2.10: Scenario of Traffic Jam

The newly developed Mashaer rail line is a rapid rail transit system that connects the Holy sites of Mina, Muzdalifah and Arafat; it is aimed at drastically reducing traffic congestion at the Holy Sites. It is designed to help accommodate the continuously growing number of pilgrims and to improve their comfort. In 2010, the year of its opening, it operated at about 35% of the full capacity but already replacing about 4000 buses previously used to transport about 150000 pilgrims. The line and the comprised stations are involved in very different types of transport movements in different days of the Hajj. The line includes 9 stations: 3 in Mina (the first of which is called Jamarat), 3 in Muzdalifah and 3 in Arafat as picture 3 depicts. The rail system will remain in operation all year round and it will be used to access the Haram and Makkah Central Area.

The flows of pilgrims to and from the stations have to be organized to reduce congestion:some parts of these places represent constrained spaces, that can contain only a certain number of persons, as opposed to other areas (e.g. the Arafat plain) that can accommodate more easily a large number of pilgrims. To this end, the area around the stations must be organized (waiting areas, access control, emergency routes, etc.), to organize the flow of pilgrims in order to prevent the arrival of additional pedestrians in the constrained areas, but allowing the pilgrims to stand and wait in other nearby areas that can accommodate them safely. It is important that the personnel responsible for pilgrim guidance and access control is aware of the different types of “movements” and cooperates with station managers.

The Arafat I station is the farthest from central Makkah in the whole Mashaer line; the station lies very close to the southwestern border of the Arafat plain. The area around the station is divided into blocks and lots, which are serviced with car parks on both platform sides. The campsite between the station and Mount Arafat is very structured, while the other side faces an area on the border of the Arafat plain and it is not characterized by an equally dense presence of tents and other accommodation structures. Pilgrims enter the station by means of ramps, elevators and escalators (the latter were not operational during the 2010 Hajj). Also footbridge access have been created: these bridges pass under the platform but above the road that lies under the station and they allow pilgrims to move from one side of the station to the other. This is particularly important during the movement in which all pilgrims must travel from Arafat to Muzdalifah, since both of the platforms must be used to assure an efficient transportation of pilgrims.

One of the most demanding movements that the infrastructure of the Mashaer Rail line must be able to sustain is the one that takes place after the sunset of the second day of the pilgrimage, which involves the transport of pilgrims from Arafat to Muzdalifah. The pilgrims that employ the train to proceed to the next phase of the process must be able to move from the tents or other accommodation to the station in an organized flow that should be consistent with the movement of trains from Arafat to Muzdalifah stations. Since pilgrims must leave the Arafat area before midnight, the trains must continuously load pilgrims at Arafat, carry them to Muzdalifah, and come back empty to transport other pilgrims. Trains move in a coordinated process to assure a consistent flow of pilgrims: when a train loaded with pilgrims leaves the Arafat I station heading to Muzdalifah I on one of the lines, on the other line a different train moves back empty to allow other pilgrims boarding it on the other platform. Since the western side of the station is far from the tent and accommodation area, pilgrims mostly reach it by means of the previously introduced footbridge access.

The size of the platforms was determined to allow hosting in a safe and comfortable way a number of pilgrims also exceeding the potential number of passengers of a whole train. Each train is made up of 12 wagons, each able to carry 250 passengers for a total of approximately 3000 persons.

In order to achieve an organized and manageable flow of people from outside the station area to the platforms the departure process was structured around the idea of waiting–boxes: pilgrims are subdivided into groups of about 250 persons that are led by specific leaders (generally carrying a pole with signs supporting group identification by pilgrims). The groups start from the tents area and flow into these fenced queuing areas located in immediately outside the station, between the access ramps. Groups of pilgrims wait in these areas for an authorization by the station agents to move towards the ramps or elevators. In this way it is possible to stop the flow of pilgrims whenever the number of persons on the platforms (or on their way to reach it is using the ramps or elevators)is equal to the train capacity, supporting thus a smooth boarding operation. The planning of group arrival at the station but also the coordination of group leaders with the station managers and other managing officers, is crucial to assure a safe, smooth and comfortable overall process of departure from Arafat for the pilgrims performing the Hajj.

Arafat 1 is a modern station but due to its position and role in the context of the Hajj rituals some particularities have been taken into account and they caused some particular design choices.

First, of all Arafat 1 it is an elevated station and has many types of access.

It has been designed to be a rapid transit station, so that no kind of service is hosted in it. The observation of the station has highlighted some aspects which could not be foreseen in the design phase. The really hot climate of Saudi Arabia produce some unexpected distribution of the pilgrims outside the station. This distribution appears to be non homogeneous and concentrates on the shady areas offered by the structure of the station and by the sparse trees.

The positioning of the waiting boxes, with respect to the close tent areas and related roads should be considered, especially given the probable increased usage of the station during the next Hajj. Groups of pilgrims coming from the tents area could find relatively narrow the passage towards the waiting boxes, especially if the groups arrival plan is characterized by a schedule that is too dense or in case of early/late arrivals due to errors of group leaders. During the observations carried out in the context of Hajj 2010, in one case a group moved directly from the tents area towards one of the ramps. At the same time, another groups from a waiting area was already approaching the ramp. This conflict caused a longer than average waiting time of other groups, due to the fact that more pilgrims than usual were climbing the ramp (or waiting to do that).

The Arafat I station does not provide facilities for pilgrims like toilets, chairs and so on. The rationale of this design choice is to support relatively quick boarding and alighting operations, but it could be problematic in case of long waiting times for elderly people. According to observation carried out during the Hajj in 2010 it was possible to identify situations in which pilgrims were looking for places and objects to sit down, like short walls, chairs for the station personnel and so on. On the other hand, it is apparent how in case of saturation of the alighting/waiting or boarding area on the station platform the presence of such kind of facilities could represent an obstacle to the flow of pilgrims.

Other considerations could be made on the access ways to the station from outside: these are mainly ramps and elevators. During the observations, elevators were not reserved to groups including people with disabilities or walking problems, like aged people: this leads to longer waiting times that could represent an issue in very hot days. Moreover it must be noticed that the future construction of escalators will represent a serious improvement simplifying the management of people having problems in climbing the ramp, whose length and slope represent a problem for people with disability or walking problems.

Chapter 3

State of the Art

3.1 Crowd Dynamics: Integrated Approach

Crowd studies represent successful applications of researches carried out in the context of computer simulation and computer vision. In fact, comprehensive studies require the synthesis of pedestrians and crowd behaviour but the developed models must be (i) properly calibrated and validated by means of data acquired on the field and (ii) informed by the specific contextual conditions of the simulated environment (e.g. number and positions of pedestrians in the area). Synthesis requires thus the results of analysis. In turn, the analysis of crowding phenomena can benefit from results on the side of synthesis: researches on the latter often produce formalization of phenomena, lead to the definition of metrics and indicators to evaluate the generated dynamics. These concepts and mechanisms can represent a useful contribution towards the automation of the analysis techniques that, thanks to the development of computer vision techniques, can actually produce useful information even from cluttered scenes like those taken from security cameras in public spaces. The overall resulting cycle of integrated synthesis and analysis of pedestrian and crowd dynamics is depicted in Figure 3.1.

Since the aim of this thesis is to understand pedestrian and crowd dynamics by employing computer vision technology, therefore we will focus on the details of different techniques aim at solving different problems of crowds using computer vision. Computer vision researches have produced significant results on the automated analysis of pedestrian and crowd behaviour, but before providing a detailed discussion of the results, we provide a qualitative classification of crowds that considers high and low density situations, as shown in Figure 3.2, which was inspired by [1].

Although there is not a common approach that can differentiate high density crowds from low density crowds, from the analysis point of view, in low density situations there is a clear visibility of individuals with little occlusions; in these situations we can detect and track the individuals in the scene. This,

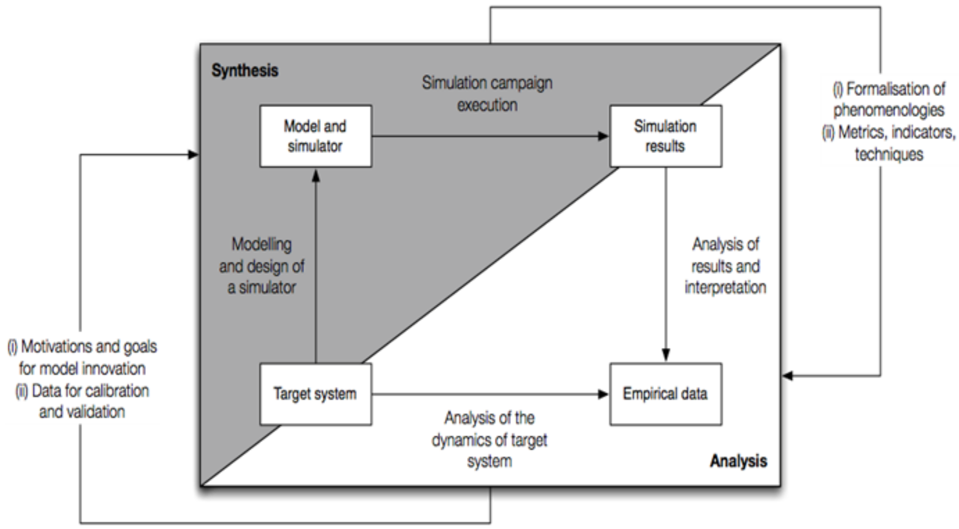

Synthesis
Simulation campaign execution
Model and simulator
Simulation results
(i) Formalisation of phenomenologies
(ii) Metrics, indicators, techniques
Modelling and design of a simulator
Analysis of results and interpretation
(i) Motivations and goals for model innovation
(ii) Data for calibration and validation
Target system
Empirical data
Analysis of the dynamics of target system
Analysis

Figure 3.1: An integrated cycle of pedestrian and crowd dynamics synthesis and analysis

on the other hand, is not usually possible in high density crowds in which detection and tracking of individuals is very challenging for a number of reasons: (i) with increasing density, the number of pixels per individual decreases; (ii) severe occlusions result in the loss of observation of the target individual; (iii) discerning individuals from one another is hindered by the constant interaction among individuals in a crowd.

High density crowded scenes, in turn, can be divided into categories, structured and unstructured [2], with some examples shown in Figure 3.3. In structured crowds, pedestrians moves coherently in a common directions that motion does not change over time. For example, the pedestrians doing Tawaf in Hajj, or pedestrians participating in marathon race, represent structured crowds because the direction of motion is fixed and does not change over time. The motion of the pedestrians in unstructured crowds appears to be not immediately characterizable, with flows of pedestrians changing direction with the passage of time. Road crossings, railway stations, expos, airports are some examples of unstructured crowds.

A survey about the crowd analysis methods employed in computer vision is presented in [3]. An interdisciplinary framework for crowd analysis to improve simulation models of pedestrian flows is also presented in [4]. The research on crowd analysis can be classified on the basis of the specific problem that reseachers are trying to solve. The researchers adopted combination of different traditional techniques or developed new techniques in order to solve different crowd problems. We categorise these problems as follows:

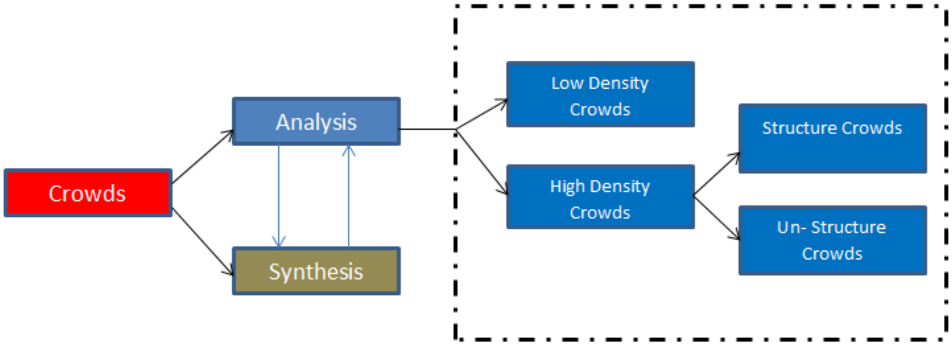


Figure 3.2: A classification of crowd studies with particular attention to analysis

1. Crowd counting and crowd density estimation,
2. Detection of individuals in crowd
3. Crowd flow segmentation,
4. Crowd behaviour understanding ,
5. Social groups in crowds.

The relevant literature of each of these categories is discussed below in details.

3.1.1 Crowd Counting and Crowd Density Estimation

The goal of this problem is to develop algorithms that can automatically localize the individuals in the crowded images and videos. Most of the research in this areas focus on estimating crowd density estimation. Crowd density is estimated by employing either segmentation of people or head counts, or by texture analysis or wavelet descriptors. Crowd counting or density estimating algorithms are generally classified into two groups: $holistic approach$ and $local approach$. In holistic approach, global features of image,i.e, textures, edges, foreground pixels are extracted from the image or video sequence and a classifier or regression model is then employed to map between the extracted feature space and the actual crowd size. In contrary, local approach utilizes the local features of image which are specific to individuals or group of people. These groups of people are independently analysed and the total crowd estimate is the sum of its parts.

Holistic approaches

Holistic approaches estimate the crowd size by utilizing the global image features. Features used by these methods include textures [5], foreground pixles [6]

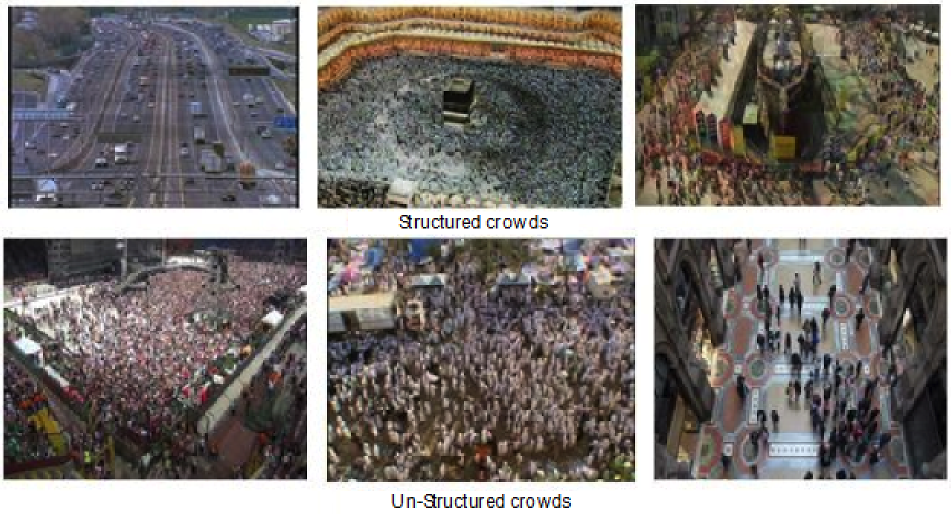


Figure 3.3: Examples of structured and unstructured crowds

and edge features [7]. The methods proposed in [5, 8] utilizes gray level cooccurrence matrix (GLCM) for crowd density estimation. Minkowski fractal dimension also proposed in [9] for extracting texture featuers. [10] showed classification accuracy of 95% when crowd density is classified into four classes by using wavelet descriptors. Classification is done by support vector machine. Their method is good for estimation of crowd density for moderate crowd density. [11] proposed Translation Invariant Orthonormal Chebyshev Moments (TIOCM). It is observed in [11] that superior performance is achieved on afternoon dataset, because the due to less variations in illumination when compared to the morning dataset. This highlights the limitation of texture features when employed in real time situations. Other approaches utilises foreground pixels and edges to estimate the crowd size. [12] proposed a number of edge features, such as vertical edges for detecting legs and arms of the individuals. [6] found the relationship between the foreground pixels and crowd size and establish a principle that number of people are linearly proportional to number of foreground pixels and that of number of edges. These features are also used by [13, 14, 15] and crowd size is estimated by employing a fast training algorithm for feedforward neural network. The mentioned approaches relied on the static background with scenes relatively at high camera angle. The total number of foreground pixels is less likely to be a good indicator because the objects in the distance appear smaller and will contribute less pixels to the foreground. Therefore as a solution [16] argues that the perspective distortions in images for pixel based crowd estimation are either incorrect or not done well, they propose a geometric correction technique, and they argue that the correction depends on y-axis only. Hence if a human is standing upright, pixels on his feet have a scale, and all the pixels on his body has the same scale as his distance from the camera is same. They use a simple foreground pixel detection technique using some masks and adaptive area growing as well. They integrate the GC into their pixel count using a lookup table. They assume each person as a rectangle changing in size with y value, and then consider all positive pixels in that rectangle as that person. The authors point out many flaws in past research works but this approach may fail when dealing with high crown density when people occlude others partially and completely. [17]uses the fore ground pixels and finds them using a Median Background computing technique. Foreground pixels are found by applying a threshold and then morphological operations are done to smooth the results. They ignore zones by masking area that have motion but not interesting like road (cars) etc. They apply classification algorithms like SVM, k-nearest, PNN, BPNN to classify the images in 2 categories first, zero persons and one and more persons. On more than zero people's categories it again applies the classification techniques to find the number of people in the scene. They train these classifiers on 70% of the images and test them on the 30% of the remaining images. The median filters are applied on the sequence of image results to get rid of the spiky errors. Also they use assorted grid to see if the accuracy increases. A very large number of features are extracted from each image by employing holistic algorithm proposed in [18, 19, 20, 21] to account for occlusion and other non-linearities.

In summary, holistic approaches tend to estimate the crowd size by exploiting global features of image. However, due to high variations in crowd behaviours, distribution and density, crowd size is difficult to estimate. Therefore,as a solution, local approaches are proposed to overcome the limitations of global approaches.

Local approaches

Local approaches use head detectors or features that are associated to individuals or group of individuals with an image. These approaches are categorised into two: i) **Detection based** approaches use head, face etc to localise the individual in an image. Once the localization is done, then crowd counting is performed. ii) **Localisation based** method divide image into overlapping blocks and then features are extracted from each block and then crowd is estimated by applying regression model.

Detection based approaches are suitable to the scenes where the crowd is spare, i.e, the people in the scene are well separated and their bodies are fully visible. Therefore human detector or head detector is employed to get the crowd count. Detection based crowd counting approaches are reported in [22, 23, 24]. A survey of pedestrian detection methods are reported in citedollar2012pedestrian,enzweiler2009monocular. Since in the real time environment, pedestrians are always occluded and their bodies are not visible enough that can be detected by pedestrian detection methods. Therefore as alternative, localisation based methods are proposed in which image is divided into number of subregions and counting is done in each region by employing regression.

Group localisation is performed by employing key points clustering method in [25, 26, 26, 26]. In this approach, SURF features were used to detect key-point in an image. Stationary points are removed by taking the mask of features points with optical flow. The remaining points are clustered int groups by using K-means algorithm. The group size is then estimated. The shortcoming of these approaches is that they are restricted to moving objects and could not count the people who are stationary in the scene. A number of approaches is proposed in literature [27, 28, 29, 30] that divide the image in sub-regions and classify discrete density level.

3.1.2 Detection of individuals in crowd

The goal of this task is to develop algorithms that can automatically localise the individuals in the scene. A number of different approaches to solve this problem are reported in literature [31, 32, 33, 34]. In [31], segmentation scheme for localizing the people is proposed. The problem was modelled in Bayesian framework where each person was localized by maximizing the posterior probability with foreground blobs. In [32], the proposed method localize the individuals by using part based detectors of edge features. Detecting individuals using interest points has also been proposed in literature. [33] proposed a global annealing framework for localizing individuals in crowd using clustering of interest points based on the geometric association with each other. [34] proposed a Bayesian clustering method to group trajectories on basis of space-time proximity. Simple image features are tracked and group them probabilistically into clusters that represent independent moving persons.

Stero-based head detection approach is reported in [35]. The algorithm is based on the notion that in public places like airports, railways, stations etc, the camera is mounted at large angle, isolating each individuals head from one another. The proposed algorithm is based on three stages: i), an adaptive filtering is performed to extract head like objects; ii), a perspective correction is then performed and iii), a mean-sift used to locate human heads in likelihood map. [36] proposed a different method based on Haar-Wavelet feature for head detection. Their algorithm is based on notion that heads of pedestrians form a texture which can be distinguished from the background using wavelet features. An algorithm reported in [37] that detect individuals from a foreground blob after applying background subtraction. In crowded scenes, such foreground blob contain more than one person since in many cases they are occluded. The algorithm used Fourier descriptor for explaining the shape. [38] localize the objects by using multiple lasers. The background modelling of image scan by laser was used to detect foreground blobs. Similar work by employing a laser scanner at the ground level for detection and tracking is reported in [39, 40].

The main limitation of these methods is that these methods are applicable to low density situations, where the people are sparsely distributed in the environment. But these approaches become impractical when applied to high density situations, where the people are highly occluded and have less pixels per person.

3.1.3 Crowd Flow Segmentation

In video surveillance,“detection and tracking” are the core technologies but these technologies are likely to fail in high density crowded scenarios. An important contribution that automated analysis tools can give to pedestrians and crowd safety is the detection of conflicting large pedestrian flows: this kind of movement pattern, in fact, may lead to dangerous situations and potential threats to pedestrians’ safety. Therefore,segmenting typical flow patterns of crowd and estimating the number of people in crowd are important steps to understand overall crowd dynamics. Most algorithms developed for object detection and tracking work well with pedestrians in low density crowds where the number of people is generally less than twenty individuals in a single frame, but with higher densities (where the number of people in a frame can be in the order of hundreds), detection and tracking of individuals are almost impossible due to multiple occlusions.

Therefore, the research has focused on gathering *global motion information* at higher scale. Global analysis of dense group of moving people is often based on *optical flow analysis*. [41] proposed particle dynamic segmentation of crowd flows by detecting the lagrangian coherent structures over the phase space. But their proposed method is computationally expensive because of the calculation of FTLE and also could not detect small flows. [42] used SIFT features to detect dominant motion flows. Flow vectors of SIFT features are calculated and then motion flow map is divided into small regions of equal size. In each region, dominant motion flows are estimated by clustering flow vectors. [43] proposed spectral clustering method for crowd flow segmentation by computing sparse optical flow field. Crowd flow is estimated using multiple visual features reported by [44] where flow is estimated by the number of persons passing through a virtual trip wire and accumulate the total number of foreground pixels. Min-cut/max flow algorithm is used by Ullah et al. [45] for crowd flow segmentation. In all of above four methods, we can not find clear boundaries among different flows. Crowd flow segmentation by using histogram curves is reported by [46] where angle matrix of foreground pixels is segmented instead of optical flow foreground. The derivative curve of histogram is used to segment the flow. Since this method only looks to the peaks of histogram curve, therefore it loses information about the crowd flow.

The above mentioned technique segments the crowds flows into different segments depending on the basis of different directions. Such analysis are very useful in detecting and predicting conflicting flows but these analysis lack information about the start or stop location of different flow segments. Therefore, in the next sub-section we discuss techniques that can automatically detect source and sink points of different complex flows in the scene. These analysis help us in crowd behaviour understanding and also have many applications for the researchers working on crowd synthesis.

3.1.4 Crowd Behaviour Understanding

Traditionally, crowd analysis is performed by the analyst who manually identifies and detects different relevant behaviours in the scene. A portion of video is given to each analyst together with a list of events (behaviours) and objects to look for. The analyst informs the concerned authorities if any of the given events or objects are detected. Such kind of manual analysis of video is labour intensive, time consuming and prone to errors due to weak perceptive capabilities of humans, but also to the repetitiveness of the activity.

In video surveillance, scene modelling and understanding is also an important research area. Important tasks of scene modelling and understanding are (i) extracting motion information (e.g. trajectories), (ii) identification of entry and exit points of trajectories in the analyzed scene, (iii) characterization of the interaction of trajectories (highlighting, for instance, crossings or potential conflicts).

With the advancement in computer vision technology, researchers developed tracking methods that in certain conditions can automatically detect, track and identify specific activities in the scene. [47] developed a tracking algorithm by modeling human shape and appearance as articulated ellipsoids and color histogram respectively for crowded scenes. [48] use Markov chain Monte Carlo based particle filter to handle interaction between multiple targets in crowded scene. [49] detects interest points in each frame by tracking pedestrians, and this activity is performed by finding correspondence among points between frames. [34] developed an unsupervised bayesian clustering method to detect individuals in crowd: for each frame, detection of individuals is performed ignoring the relationship between frames. [50] detect individual objects on the basis of assumption that objects move in different directions. [51] develop a tracking system by solving data association problem by utilizing Generalized Minimum Clique Graph (GMCP) in order to detect an individual in different frames of a video. Intuitively, detection and tracking of individuals rely on the performance of detection and tracking algorithms. However, in crowded scenes, where the number of objects is often in the order of hundreds, these tasks usually fail due to (i) the variable and potentially low number of pixels per object and (ii) frequent and severe occlusions related to the constant interaction among the objects (pedestrians) in the scene. These challenging characteristics of the analyzed videos can be at least partly avoided in laboratory situations: for instance, in [52] the authors successfully gather pedestrians' trajectories and gather useful data about their behavior but they employ a manual or automatic but facilitated form of identification. Moreover, as we will discuss in the experimental evaluation that the adopted tracking algorithm (Lucas-Kanade tracker - KLT [53]) does not provide sufficiently accurate results in naturalistic conditions.

Intuitively, detecting sources and sinks (as introduced above) implies detection and tracking of objects, potentially followed by an analysis of the trajectories: this kind of approach was adopted by [54], which analyses low density situations and essentially relies on the performance of the tracking algorithm, which is low in crowded situations. Research in this area has therefore instead assumed that raw data about pedestrian paths should be considered as noisy or unreliable: [55], for instance, employ a so-called *weak tracking* system and they aggregate raw *tracklets* through a mean-shift clustering technique allowing them to identify entry and exit zones in the scene. More recently, in order to overcome the limitation of traditional tracking methods, research has focused on gathering global motion information at higher level, often based on optical flow analysis.

Trajectories capture the local motion information of the video. Long and dense trajectories (that is, trajectories representing a large number of paths followed by different pedestrians, reaching a significant length) provide good coverage of foreground motion as well as of the surrounding context. There are two types of representations for characterizing motion information from the video: space-time local features (like corner points, SIFT features etc.) and dense optical flow [56]. In the first type, features are detected in one frame which are then tracked through rest of the frames of a video, whereas the second type is based on dense optical flow, where a flow vector is estimated for every pixel. Since dense optical flow estimates a change for every pixel it provides a better representation of motion in video. A large number of approaches for extracting feature trajectories from video exist:

- the work described in [57] extracts feature trajectories by tracking Harris3D interest points; [58] used KLT for extracting trajectories represented as a sequence of log-polar quantized velocities which later on used for action classification;

- a different approach [59] also used KLT for extracting trajectories, that are then clustered and affine transformation matrix representing trajectories is computed for each cluster;

- other researchers extract trajectories by matching SIFT descriptors between two consecutive frames [60];

- the work described in [61] combine both KLT tracker and SIFT descriptor matching to extract long-duration trajectories, and random points are sampled for tracking within the region of existing trajectories in order to assure dense coverage;

- another approach [62] extracts feature point trajectories in the regions of interest; in this work, authors compute histogram of gradient (HOG) and histogram of optical flow (HOF) descriptors along the trajectories.

- KLT method is also used in [63] for extracting sparse trajectories: the authors propose Random Topic Model (RTM) for learning semantic regions from the motions of pedestrians in crowds. A variant of this approach [64] employs KLT trajectories and proposes Mixture model of Dynamic pedestrian Agents (MDA) that analyse the collective behavior of pedestrian in crowds after learning from the real data.

Resulting trajectories from the above approaches are effectively long duration but they are typically sparse and can not capture whole motion information of the video because only few feature points are detected.

On the other hand, dense optical flow captures whole motion information of the video, as we estimate a flow vector for every pixel of a frame; but due to the unpredictable nature of the pixel (due to its sensitivity to the illumination), we can not extract reliable long duration trajectories. There is limited literature about dense trajectories:

- an approach described in [65] extracts long range trajectories using dense optical flow;

- a different approach [66] extracts objects from video using dense optical flow trajectories;

- video is represented as set of particles and their trajectories are computed using variational optical flow in [56];

- in [67], particle trajectories are obtained by overlying a grid of particles on the initial frame of video, initializing a dynamical system. Time integration of this dynamical system provides particle trajectories that represent motion in the scene. This method represents a very useful starting point for our goals, especially for generating robust local movement trajectories, although it is not aimed at providing global pedestrian motion information but just for identifying specific crowding situation or movement patterns.

Generally, these techniques are quite reliable when so called *structured crowds* [68] are analyzed: this is mostly due to the nature of this kind of situations, when flows of pedestrians can include a very large number of individuals that, however, follow relatively stable flows that are generally well separated and not conflicting (e.g. people in a marathon, pilgrims performing Tawaf during the Hajj). Achieved particle trajectories in high density unstructured crowds are, instead, normally not accurate and unreliable due to (1) severe occlusions that occur frequently, (2) ambiguities arising at the boundary of the conflicting flows as reported in [69]. In these cases, a particle can drift to the side of another motion boundary and it can mix with different motion. Other approaches like, [70] and [71], which extract motion trajectories using KLT and adopt hierarchical clustering algorithms for detecting dominant flows in scene. These methods do not consider the whole video, but they rather consider a portion of it; moreover they do not actually try to identify sources and sinks of different flows but rather capture information about a low number of frames which provide inadequate information for understanding the overall behavior of the scene. In this thesis, we adopted a similar approach as [67] for extracting motion information, but we overcome the limitations of the previous approaches by employing rules for extracting highly accurate and reliable particle trajectories.

3.1.5 Social Groups Detection in Crowds

In thesis, we investigate another important problem of crowd dynamics,i.e, social group detection. As we understand from our knowledge that crowded scenes are composed of large number of people exhibiting different behaviors in a constrained environment. The analysis of the behavior of pedestrians and crowds in video surveillance systems is a topic of growing interest supporting an improved understanding of human behavior and decision making activities through several functions like activity recognition [72], automated analysis of the flow of large crowds, for example through crowd flow segmentation and crowd counting [73], the discovery of frequent pathways [74], the identification of crowd behaviors [67] and abnormal event detection [75, 76]. All these studies either focus on individuals or on the overall crowd, considered as large set of pedestrians, not considering the importance of some social interaction among pedestrians: most pedestrians do not really walk alone [77], and researchers observed in most situations pedestrians actually walk in groups. Some interesting forms of social interaction and adaptive behaviors can be observed at the group level and they are growingly investigated in the area of pedestrian and crowd modeling and simulation [78, 77]. On the other hand, detecting and analyzing social groups of people is still a less studied topic.

A few recent works [79, 80] are aimed at the detection of groups without using future information about the dynamics of the scene. [79] employed Decentralized Particle Filtering (DPF) for group detection while [80] employed unsupervised group detection method based on Dirichlet Process Mixture Model (DPMM) which exploits proxemics to determine group formation. Other approaches like [81, 82, 83] use social forces to analyze motion patterns and recognize groups. These social forces based methods are based on pairwise similarity between trajectories of pedestrians followed by a clustering phase. An approach described in [84] extracts trajectory information from the whole video, then trajectories are temporally analyzed in order to determine the affiliation of each pedestrian to a particular group. Pedestrians are grouped in a bottom-up fashion by employing hierarchical clustering using pairwise proximity and velocity. In [85], both spatial locations and velocities are used within a modified Hausdorff distance to compute trajectory similarities. In [86], Euclidean distance metric is used to cluster vehicle trajectories. [74] measures trajectory similarities using Longest Common Sub-Sequence. [87, 88] use Hausdroff and Dynamic Time Warping metric to measure trajectory similarities. The problems with employing all above pairwise similarity measures are that they are computational expensive and lack probabilistic explanation. On the other hand, instead, recent works are focusing on modeling the distribution of trajectories locations and velocity observations [89, 90].

Chapter 4

Crowd Flow Segmentation and Crowd Counting

4.1 Overview

As the population of world is increasing and ever more located in urban areas, public safety is becoming a problem in most crowded areas of the big cities. Mass events like those related to sports, festivals, concerts, and carnivals attract thousands of people in constrained environments, therefore adequate safety measures must be adopted. Despite all safety measures, crowd disasters still occur frequently. The reasons of these disasters is mostly the presence of different and conflicting motion patterns that influence the crowd. A crowd is composed of small groups of people, for instance due to social relationships (families or friends) or a common goals, like reaching a certain point of the environment. The latter groups can be called short term coherent groups because they discontinue their cohesion after completing the goals (e.g. reaching an exit, completing a movement). Detecting the second kind of group, essentially associated to a certain flow of pedestrians in the environment, can be important to be able to prevent conflict situations.

Due to the complex dynamics of the crowd, crowd management is becoming a daunting job where huge effort from the security staff is required to manage the potentially problematic situations. In such high density crowded areas, surveillance cameras are generally installed in different locations that can even cover the whole scene. Detecting specific activities in real-time videos is the task of analysts sitting in surveillance room and watching over multiple Tv screens. Such manual analysis of high density crowds is a tedious job and usually prone to errors. Therefore we need automatic analysis of the crowd which can reliably estimate the density and detect specific activities. Creating such kind of virtual analyst has become the focus of many researchers. This research has a wide range of application domain in crowd management, public space design, underwater fishes analysis (and animal behaviour studies in general), and cell

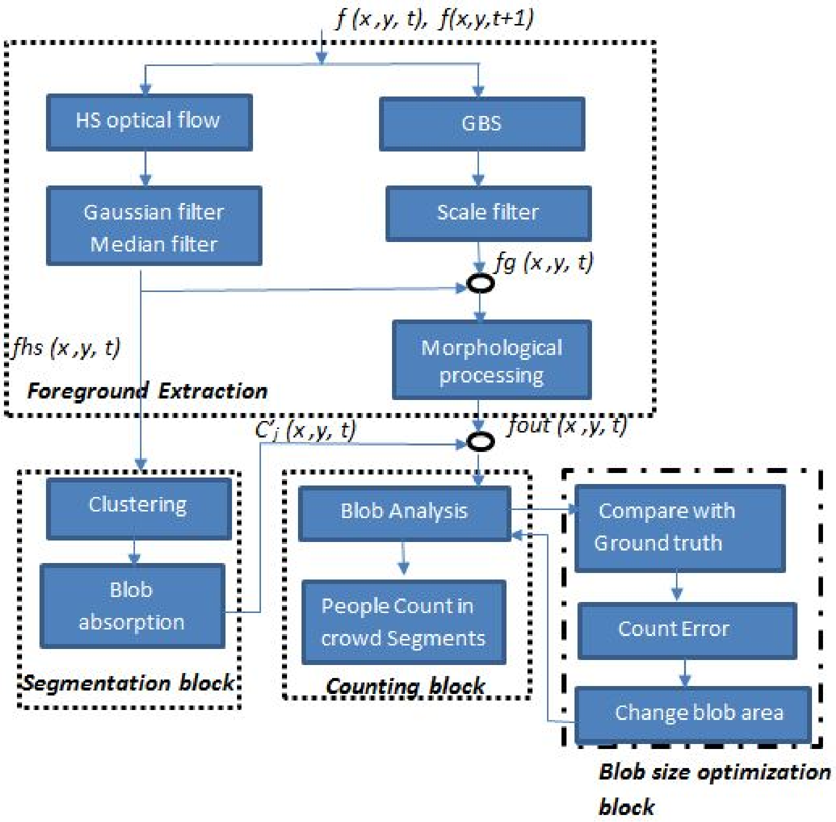


Figure 4.1: Overview of proposed framework

population analysis. In video surveillance, "detection and tracking" are the core technologies but these technologies are likely to fail in high density crowded scenarios. In this paper, we propose a framework that tackles problems of crowd flow segmentation, crowd counting and consists of three parts: foreground extraction, crowd flow segmentation, and crowd counting. In the next section i discuss the framework for crowd flow segmentation and crowd counting.

4.2 Proposed Framework

Our proposed framework is composed of four processing blocks, Foreground extraction, segmentation, counting and blob size optimization block, but this block only executes in the beginning for few initial frames. In this section, we will discuss each processing block in detail. For sake of description of the proposed approach we will employ videos taken from a crowd related data set from UCF [41].

4.2.1 Foreground Extraction

Foreground extraction is the most important pre-processing step for detecting the moving objects from the video and therefore forms the basis of our framework. Foreground extraction is useful for detection, tracking and understanding the behavior of the object. A survey on motion detection techniques can be found in [91]. Traditionally, in video surveillance with a fixed camera, researchers use background subtraction method, where foreground objects are extracted from video if the pixels in the current frame deviate significantly from the background. In this paper, we use two foreground masks as in [92], one generated by optical flow, $f_{hs(x,y,t)}$ and will be used by crowd flow segmentation framework and other is Gaussian background subtraction, $f_{g(x,y,t)}$ used by counting framework as shown in Figure 5.1. Two consecutive frames $f_{(x,y,t)}$ and $f_{(x,y,t+1)}$ are applied to foreground extraction block. First, we compute Horn and Schunk (HS from now on) optical flow between adjacent frames, then Median filter and Gaussian filter are used to remove noises. We then set a threshold to get foreground mask $f_{hs(x,y,t)}$. In the same way, Gaussian Background Subtraction (GBS from now on) is used to get another foreground mask $f_{g(x,y,t)}$, after applying scale filter. Usually crowded objects move in wide areas, and for crowd flow segmentation, we need to detect change in every pixel, so optical flow methods reported in literature to compute sparse optical flow using the interest points (Lucas-Kanade optical flow) [93] or dense optical flow for all pixels (HS optical flow) [94] in each frame can be used. Since, we want to detect change in every pixel, we compute dense optical flow. Since the optical flow vector of each pixel has the magnitude and direction values, we use magnitude information to extract foreground, all the pixels which have higher magnitude than T_{th} will be classified as foreground. Direction information of optical flow vectors can be used in crowd flow segmentation by clustering all optical flow vectors having similar orientations. Such methods are usually prone to errors due to unpredictable behavior of the pixels which change due to fast/slow moving objects and illumination. A small change in illumination can be detected as foreground objects even in the static background. Such methods can be useful in extracting region of interest (ROI) in the scene but can not be used in separating individuals in high density scenarios. As shown in Figure 4.2, $f_{hs(x,y,t)}$ can not provide information about the group of foreground pixels (blobs) related to the people in the crowd. Therefore, for counting framework, we generate another foreground mask $f_{g(x,y,t)}$ by Gaussian background subtraction method. GBS is a kind of background subtraction method [95] and is very good in separating objects from the background. GBS method is effective in suppressing noise and robust to change in illumination. $f_{g(x,y,t)}$ is also a binary image, where blobs represents the objects of different sizes. Small blobs are related to parts of object, medium blobs related to objects and large blobs represent group of objects, appeared due to occlusions. Optimal foreground mask $f_{out(x,y,t)}$ is obtained by logical product of $f_{g(x,y,t)}$ and $f_{hs(x,y,t)}$. Later on, we apply morphological processes like morphological opening and closing on the binary image $f_{out(x,y,t)}$. The morphological open operation is erosion followed by dilation, eliminates

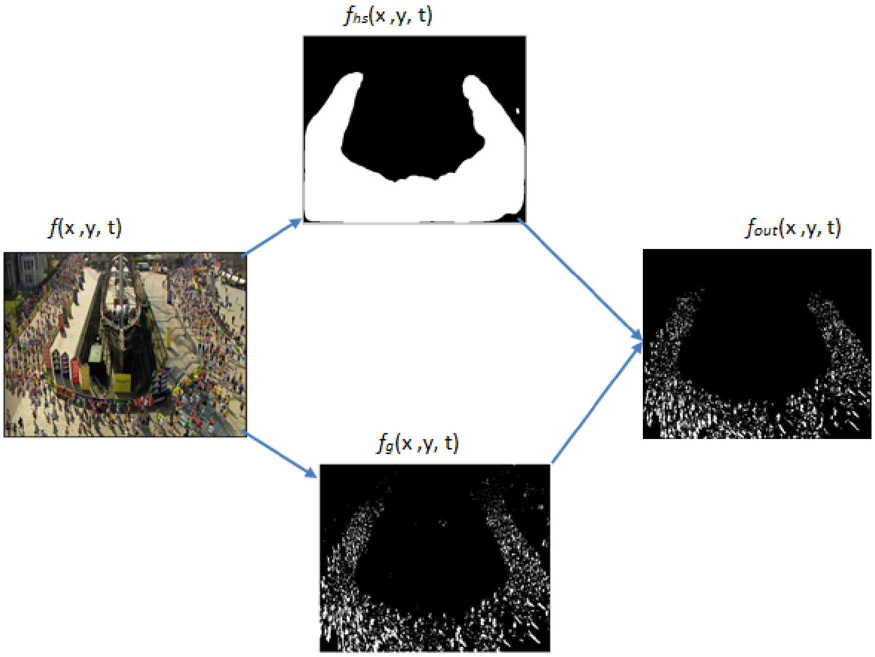


Figure 4.2: Foreground extraction framework

smooth contours and protrusions. While morphological close is dilation followed by erosion, smooths the section of contours, eliminates small holes and fills gaps in contours. These operations are dual to each other. Segmentation block segments the crowd flows into different clusters, $C'_{j(x,y,t)}$, by employing K-means clustering followed by blob absorption method. To estimate the number of people in each flow segment, we take logical product of each cluster $C'_{j(x,y,t)}$ and foreground mask $f_{out(x,y,t)}$ and count the number of people by blob analysis and blob size optimization methods.

4.2.2 Motion Flow Field Computation

After foreground extraction, the objects in the foreground move in different directions as shown in first row of Figure 4.3. It can be seen that in each video, foreground objects have multiple flows. Since we use dense HS optical flow that computes movement of every pixel, we call it motion flow field. The motion flow field is a set of independent flow vectors in each frame and each flow vector is associated with its respective spatial location. This instantaneous motion field of the video contains temporal information and can be used for the learning motion pattern of the video. Consider a feature point i in F_t, its flow vector Z_i includes its location $X_i = (x_i, y_i)$ and its velocity vector $V_i = (v_{x_i}, v_{y_i})$, i. e. $Z_i = (X_i, V_i)$ where θ_i is the angle or direction of V_i, where $0° \leq \theta \leq 360°$.Then

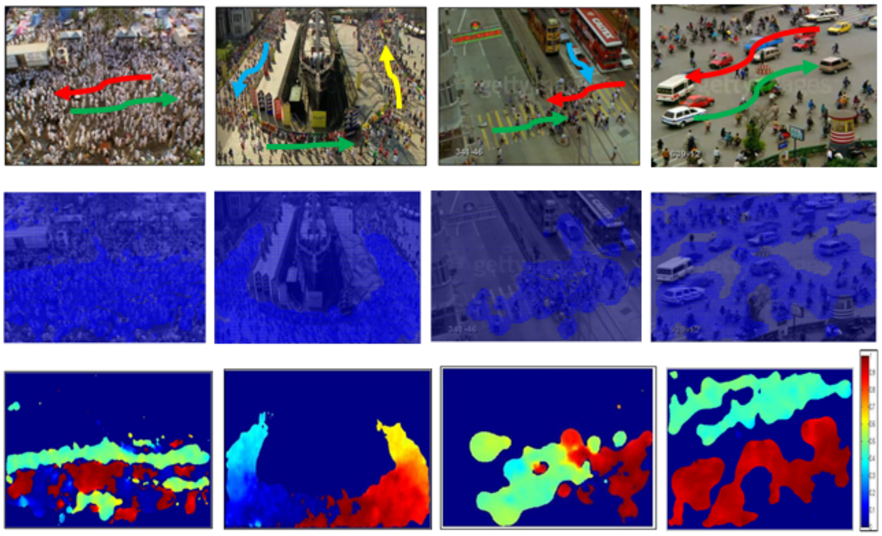


Figure 4.3: First Row: sample frames from videos of the Hajj, a marathon, pedestrian crossing, and road section; second row: corresponding optical flow; third row: corresponding direction map

$\{Z_1, Z_2, \ldots, Z_k\}$ is the motion flow field of all the foreground points of an image.

Motion Flow Field Segmentation

The motion flow field $\{Z_1, Z_2, \ldots, Z_n\}$ is a n x 4 matrix where each row represents flow vector i and columns represents its spatial location X_i and velocity vector V_i. n represents total number of flow vectors (foreground points). Each flow vector represents motion in specific direction as shown in Figure 4.3, third row. Figure 4.3, (third row) does not show dominant motion patterns, so we can not infer any meaningful information about flows. Therefore, we need a method that automatically analyses the similarity among the flow vectors and cluster them in multiple groups. We use K-means clustering algorithm(widely used in data analysis and image segmentation) to segment motion flow field into different groups. This process of grouping vectors that represent specific motion pattern is called segmentation. After segmentation process, motion field is divided into multiple segments. We denote K as the initial number of cluster centroids. Commonly used initialization methods are Forgy and Random Partition [96]. We initialize cluster centroids as $(K-1)$ x $360°/K$. let C = $\{1, 2, ..j\}$ is the set of initial cluster centroids. $\epsilon = 360°\ /2K$ and $\delta = 360°/K$.

This approach can be applied to the images where the objects moves in every direction. For such kind of complex movements in images, we assign larger value of K while we assign lower value to the images where objects move in regular directions. In this paper, we assign lower value of $K = 4$ because

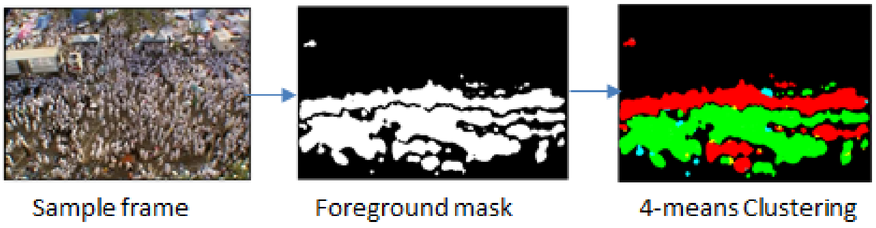


Figure 4.4: Results of 4-means clustering in a Hajj video frame

in our benchmark videos, objects move in regular directions. Figure 4.4, shows that the objects in sample frame are clustered into different groups by applying 4-means clustering. We use different colors to differentiate clusters. Let C = $\{1, 2, ..., K\}$ is the set of clusters found in sample frame.

```
Step 1 Clustering with initial K-centroids
  for $1 \leq i \leq n$ do
      for $1 \leq j \leq K$ do
          if $\| \theta_i - c_j \| \leq \epsilon$ then,where $c_j \in C$
              $z_i(x_i, v_i) \rightarrow c_j$
              $n_j \leftarrow n_j + 1$
          end if
      end for
  end for
Step 2 New centroids calculation
  for $1 \leq j \leq K$ do
      $c'_j = \sum_{i=1}^{n_j} \theta_i / n_j$, Update $C$ with new centroids $c'_j$
  end for
   Step 3 Clustering of similar clusters
  if $\| c'_l - c'_m \| \leq \delta$ then
      $c'_l = \sum_{i=1}^{n_l+n_m} \theta_i / n_l + n_m$
      $c'_m \leftarrow c'_l$
  end if
Step 4 Return to step 1
```

Blob Absorption

We noticed that after K-means clustering, some small blobs appear: these small blobs represent small clusters as shown in Figure 4.4 and resulted due to following reasons. First, if the objects move slowly, the inside and outside flow vectors of the objects are not same and as a result are classified into two different flows. Second, if the two opposite optical flow intersect, the optical flow at the boundaries is ambiguous. Third, small blobs represents small groups of people and are not the part of dominant motion flows and they are not relevant to the aims of our analysis. Therefore, we adopt blob absorption approach (mimicking a "big fish eats small fish" process), where these blobs are either absorbed by dominant cluster or by the background. The algorithm is as follows:

1. Compute weights for all clusters, i.e. $C_{wj} = \sum_{j=1}^{K} n_j \ / \ T$. where n_j is number of features points $z(x,v)$ in cluster C_j and T is total number of foreground points.

2. Select cluster C_j and perform blob analysis and find area of each blob in C_j.

3. Use threshold area L and find blobs whose area $A \leq L$. Let $B = \{b_1, b_2, ...b_n\}$ set of blobs represents small clusters and needs to be absorbed.

4. Select blob b_i from set B, find its edges points by using canny edge detector [97].

5. For each edge point, look at its neighborhood points, find neighborhood cluster ids and store ids of neighborhood points in array S. Remove those points from S that have same cluster id j, because b_i can not be absorbed by itself.

6. From remaining points in S, compute blob weight $b_{wi} = \sum_{j=1}^{N} n_j \ / \ T_s$. where N is the total number of cluster ids found in S. n_j is total number of points with cluster id j and T_s is total number of points in S.

7. Compute w_t= c_{wi} + b_{wi} and cluster id j with maximum weight wt is selected and id j is assigned to all points of blob b_j.Hence blob is absorbed.

8. Repeat steps 4 to 7 until B is empty.

9. Repeat step 2. Here background is also considered as cluster with id and cluster weight $c_w = 0$.

After blob absorption, as shown in Figure 4.5, small clusters (C_3 and C_4) are removed leaving behind large clusters (C_1 and C_2) representing dominant flows with clear boundaries, by setting up threshold area $L = 500$. Let C'=$\{1, 2, ..j\}$ is set of large clusters.

4.2.3 Counting People in High Density Crowds

This section describes the methodology for counting people in high density crowds. In this step, we count the number of people in each cluster C'_j. In low density crowds, due to clear visibility of individual with little occlusions, we can detect, track and count the number of individuals in crowd, but in high density crowds, it is hard to extract and count the individuals due to (i) with increasing density, the number of pixels/individual decreases (ii) severe occlusions result in the loss of observation of the target individual (iii) discerning individuals from one another is caused by constant interaction among individuals in a crowd. Therefore, as a solution, we perform blob analysis and blob

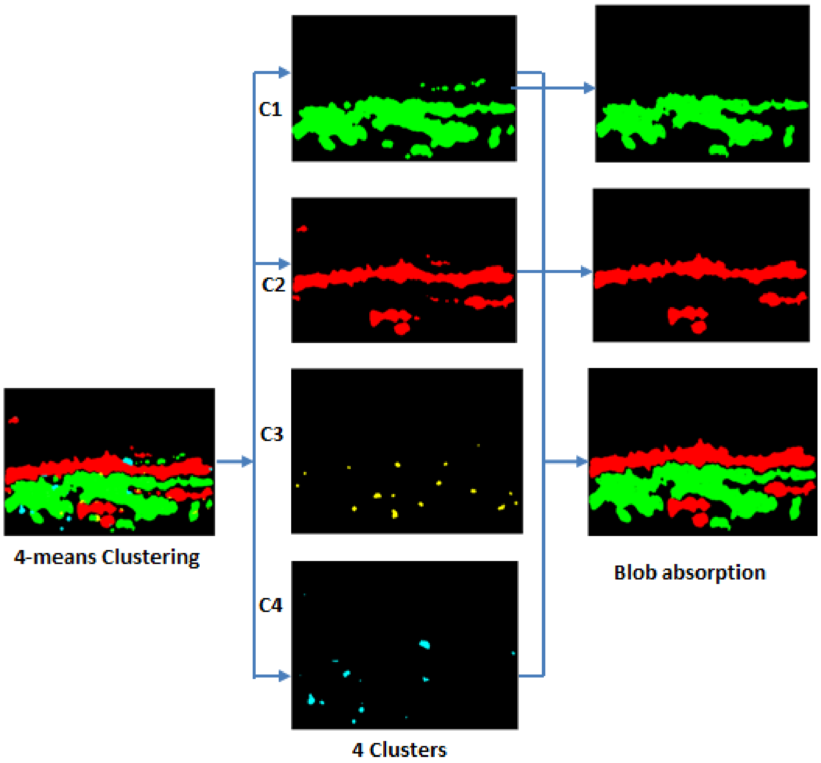


Figure 4.5: Results of the Blob Absorption method applied to a frame of the Hajj video

size optimization techniques on foreground image and estimate the number of people in high density crowds.

Blob Analysis and Blob Size Optimization

For extracting foreground, belonging to each dominant flow (or cluster C'_j), we take logical conjunction of each cluster C'_j and foreground mask $f_{g(x,y,t)}$, generated by Gaussian background subtraction and shown in Figure 4.7. First row of Figure 4.7, shows that sample frame of marathon video is segmented into three dominant flows while second row shows foreground elements belonging to each of three segments. After foreground extraction, small blobs appear which represent moving objects. Blobs are the connected regions of variables “areas” in the binary image. Since there are many blobs of different areas representing different moving objects we need to find an optimal area that will serve as a threshold. The blob with areas above this threshold will not be taken into account (for instance, when counting pedestrians in road videos, these large blobs might be related to cars). For computing threshold area we devised blob size optimization algorithm discussed below.

1. Select the blob's size randomly.lets blob's size is A.

2. c_i = blobAnalysis(A); will return count of blobs whose size $\leq A$ for frame i.

3. $error_j$ = $\parallel c_i$ - $gth_i \parallel$. where gth_i is the ground truth count for frame i.

4. Vary the blob size A by some constant k and repeat step 2 to 4 for N iterations.

5. Select blob's size A for which $error_j$ is minimum.

Note that for finding optimum blob size, we used only four or five initial frames whose ground truth is available. These frames are selected randomly. For each initial frame we compute optimum blob size by using the method discussed above. We take the mean A' of all four or five optimum sizes computed for each initial frame and use A' for counting people in rest of frames. Average and standard deviation of the error between people count using the blob area and the actual number of people (Ground Truth) is plotted in Figure 4.6 versus blob area. In Figure 4.6, mean and standard deviation of the counting error is plotted for a road video. It can be seen from the figure that the error is minimum for the blob area 17, resulting therefore context dependent. It must be stressed that the optimal blob size depends on the video, especially on the point of vantage determining the size in pixels of people to be counted (in other videos analysed in Sect. 6.4 the optimal blob size is as small as 2 pixels). Through experiments, we observed that for small blob areas, the count of people will be higher as the noise will also be counted as people. For large blob areas, instead, some people might be missed in the count. Hence selection of optimal blob size is very important to minimize the error in people count.

4.3 Experimental Results

This section presents the quantitative analysis of the results obtained from experiments. We carried out our experiments on a PC of 2.6 GHz (Core i5) with 4.0 GB memory and data set from UCF [41]. The data set covers two types of crowded scenarios: the first scenario consists of videos involving high density crowds i.e. videos from Hajj and a marathon, where the number of people is higher than 150 in a single frame. The second scenario covers low density crowds where the number of people in a frame is lower than 70, i.e. road crossing video, where people are moving over zebra crossing in different directions, and road video, where vehicles and people are moving in different directions on road. Since our framework consists of two major parts,crowd flow segmentation and crowd counting, our experiments are carried out in two steps.

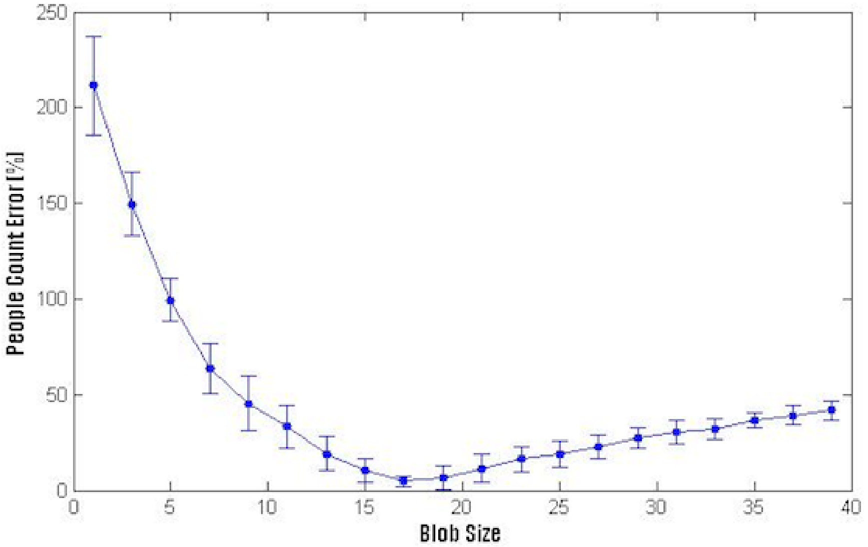


Figure 4.6: Blob size optimization for Road video: notice that the optimal blob size for error minimization is different for different videos.

Table 4.1: Hajj Video people counting in sequence of frames

F.n.	**G.T.(E)**	**G.T(W)**	**Cnt.(E)**	**Cnt.(W)**	**Err(E)**	**Err(W)**
12	151	159	170	154	12,58%	3,14%
20	153	161	167	154	9,15%	4,35%
29	185	185	195	194	5,41%	4,86%
37	176	187	192	201	9,09%	7,49%
45	187	186	200	191	6,95%	2,69%
55	187	187	195	188	4,28%	0,53%
63	189	185	194	194	2,65%	4,86%
			Average Error		**7,16%**	**3,99%**

4.3.1 Segmentation Results

We selected 65 frames from each video. After computing optical flow, we apply K-means clustering algorithm that cluster all the similar flow vectors.In this paper, we use $K = 4$ for all the videos,so after segmentation, we detect four different flows in video frame as shown in second column of Figure 4.8. We then apply blob absorption method to remove small clusters as shown in third coumn of Figure 4.8. For blob absorption we use different threshold L values. Small clusters can not be aborbed completely by using smaller values of L while we lost some portions of dominant cluster by using larger values of L. Therefore, we determined value of L experimentally and is different for different videos. After blob absorption, image of cross video is segmented into three flows, red(west),green(east) and cyan(south). While image of road video is segmented into two flows, red and green as shown in third column of Figure 4.8.

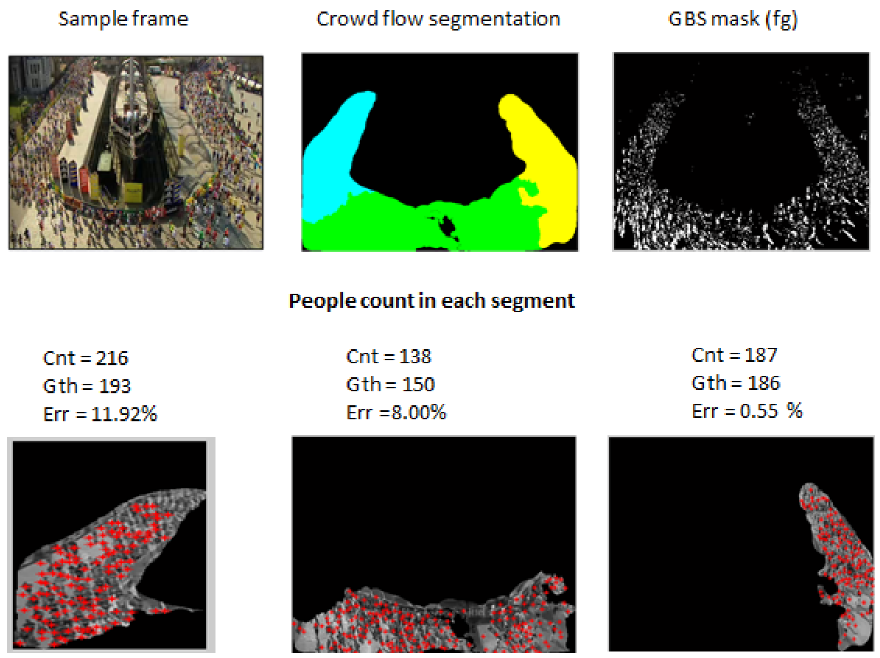


Figure 4.7: People Counting Framework highlighting results of intermediate steps in one frame of the marathon video

We compared our approach in Figure 4.9 with multi-label optimization [45], histogram curve [46], dynamic segmentation [41] and spectral clustering [43]. In the first row of Figure 4.9, we compare our method with multi-label optimization method. We see that crowd flow segmentation using multi-label optimization could not segment the crowd into dominant flows. Moreover, it could not find clear boundary due to small blobs appeared after segmentation. In the second row of Figure 4.9, we compare our results with histogram curve method. Segmentation by using histogram curve is fastest than existing methods but it lost much information about the crowd flows, since this method only looks to the peaks of histogram curves. In the third row of Figure 4.9, we compare our results with dynamic segmentation and spectral clustering approach. Dynamic segmentation is not able to detect small flows in the crowd, while spectral clustering carries out segmentation on sparse optical flow and give the approximate segmentation where we can not find clear boundaries between flows. All the above shortcomings are resloved by our proposed approach. Our proposed approach not only detects dominant flows but can also detects small flows without the loss of crowd flow information. Moreover, our proposed approach finds clear boundaries among different flows.

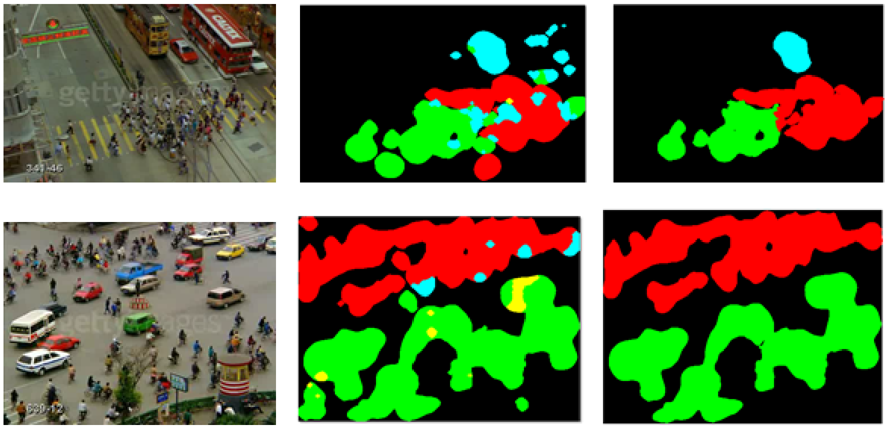


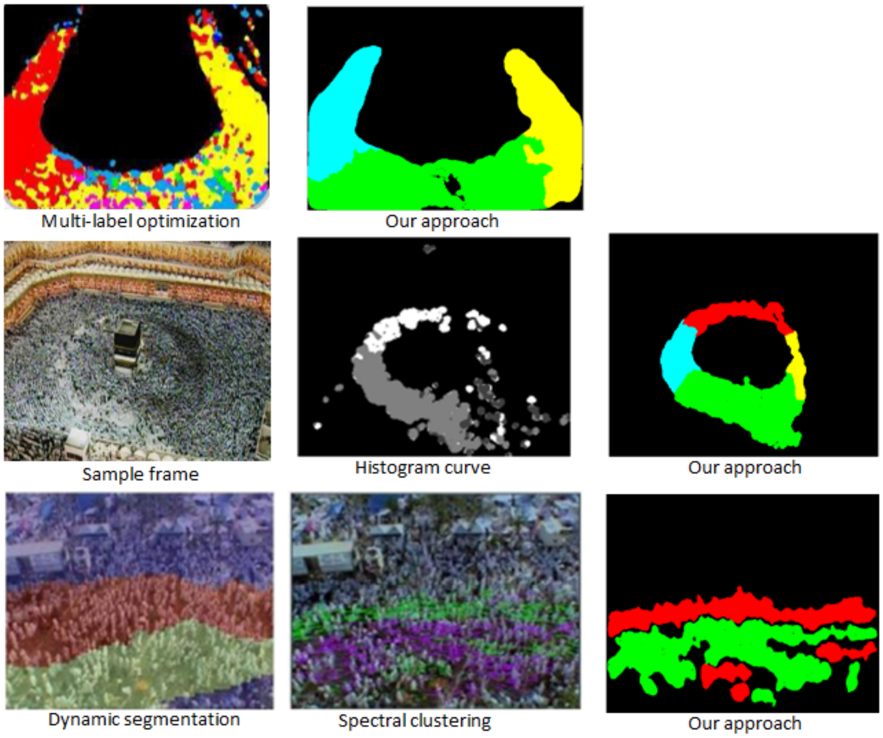
Figure 4.8: First column: sample frames; Second Column: K-means clustering results; Third Column: Blob absorption results

Table 4.2: Crossing video people counting in sequence of frames

F.n.	G.T.(E)	G.T.(W)	Cnt.(E)	Cnt.(W)	Err(E)	Err(W)
10	30	30	30	29	0,00%	3,33%
16	34	35	30	39	11,76%	11,43%
22	37	36	25	38	32,43%	5,56%
28	35	33	29	32	17,14%	3,03%
30	38	35	37	43	2,63%	22,86%
35	38	34	35	41	7,89%	20,59%
40	37	36	36	39	2,70%	8,33%
47	35	36	35	30	0,00%	16,67%
55	37	38	38	34	2,70%	10,53%
64	37	40	31	28	16,22%	30,00%
			Average Error		**9,35%**	**13,23%**

4.3.2 Crowd Counting Results

After crowd flow segmentation, we count the number of people in each flow segment. Each video consists of sequence of 65 frames and our proposed method automatically counts the number of people in each frame as shown in Tables 1, 2, 3, 4. Tables show counting results of random frames taken from each analysed video, where F.n. represents frame number of the analysed sequence. The rise and fall in people count in different frames represents the fact that people are entering or leaving the scene affecting people count at different time. To check the counting accuracy of the proposed framework, ground truth (G.T) for each direction (East(E), West(W), North(N), South(S)) is found for the frames after random intervals and count error (Err) is computed by comparing results with the ground truth data. Count error is shown in details in tables 4.1, 4.4, 4.2, 4.3 for all analyzed video sequences. The first column of each table shows the frame number, G.T. shows grouth truth found for each direction and Cnt. is counting results of our proposed approach. Average error is less than 12% for


Multi-label optimization

Our approach

Sample frame

Histogram curve

Our approach

Dynamic segmentation

Spectral clustering

Our approach

Figure 4.9: Comparing Results

all analyzed video sequences. For some frames, however, count error is higher due to the fact that some people in that frame are missed in count or noise (resulted after motion segmentation) is counted as people. As obvious from tables, our proposed framework works better in high density scenarios like Hajj and marathon. It is matter of the fact, that in high density scenarios, people covers much of the scene's area in comparison to low density scenarios. After motion segmentation, foreground extracted in high density scenarios contains less background noise (foreground noise generally moves with people and it is not causing significant errors) in comparison to foreground extracted in low density scenarios. From the experimental results, it is clear that our proposed approach count the people in each video sequence with 90% accuracy.

To study the time complexity of our proposed framework, we utilize 65 frames of each of four analysed videos and time is recorded as average frame processing time and recorded in Table 4.5. The latter shows time complexity of crowd flow segmentation and crowd counting frameworks. Rows of table shows the analysed videos and column represents time complexity of each of processing block. It is obvious that clustering takes much time as compare to blob absorption method and crowd counting framwork. It is matter of the fact that K means clustering is computationally expensive and can be very slow

Table 4.3: Road Video people counting in sequence of frames

F.n.	G.T.(E)	G.T.(W)	Cnt.(E)	Cnt.(W)	Err(E)	Err(W)
11	45	67	33	44	26,67%	34,33%
20	38	65	45	58	18,42%	10,77%
30	42	62	46	69	9,52%	11,29%
35	41	61	40	62	2,44%	1,64%
43	39	64	36	53	7,69%	17,19%
50	40	65	48	67	20,00%	3,08%
55	40	65	36	55	10,00%	15,38%
62	39	63	39	67	0,00%	6,35%
			Average Error		**11,84%**	**12,50%**

Table 4.4: Marathon Video people counting in sequence of frames

F.n.	G.T.(E)	G.T.(N)	G.T.(S)	Cnt.(E)	Cnt.(N)	Cnt.(S)	Err(E)	Err(N)	Err(S)
11	145	192	187	134	176	199	7,59%	8,33%	6,42%
15	150	186	193	138	187	216	8,00%	0,54%	11,92%
20	148	193	200	126	178	190	14,86%	7,77%	5,00%
27	155	200	211	151	244	225	2,58%	22,00%	6,64%
33	150	195	220	145	223	219	3,33%	14,36%	0,45%
39	160	205	210	151	199	222	5,63%	2,93%	5,71%
45	158	210	205	145	215	210	8,23%	2,38%	2,44%
49	156	207	210	145	189	197	7,05%	8,70%	6,19%
55	162	215	215	164	210	196	1,23%	2,33%	8,84%
59	158	220	220	162	210	202	2,53%	4,55%	8,18%
62	167	225	224	158	185	198	5,39%	17,78%	11,61%
				Average Error			**6,04%**	**8,33%**	**6,67**

to converge in worst case scenarios, i.e. high resolution videos, and high ratio of foreground to background pixels. In this paper, we use videos of the same resolution, 360x480. Although the resolution of all analysed videos is same, yet time complexity is different. The ratio of foreground to background pixels of different videos is different and usually the ratio is higher if the large part of the scene is covered by foreground pixels. It is also obvious from table that Hajj video takes more computational time than other videos. It is matter of the fact that most of scene of a Hajj video frame is covered by foreground pixels than background pixels. The computational time can be reduced and proposed framework can be employed in real time, if implemented in openCV. The current implementation is in Matlab.

Table 4.5: Time Complexity of our proposed framework in (seconds)

Videos	**Crowd Flow Segmentation**		**Crowd Counting**		
	Clustering	**Blob Absorption**	**Seg # 1**	**Seg # 2**	**Seg # 3**
Marathon	6	2.77	0.006	0.007	0.005
Hajj	9.88	2.93	0.009	0.008	NIL
Road	7.02	1.67	0.005	0.004	NIL
Crossing	5.12	1.03	0.003	0.005	NIL

Chapter 5

Crowd Behaviour Understanding: Identifying Sources and Sinks and Characterizing Main Flows

5.1 Introduction

Crowded scenes are composed of a large number of people, exhibiting different behaviors in a relatively constrained space. The vagueness of this definition is strictly related to the difficulties in defining what a crowd of pedestrian is; we will not try here to be more specific or precise, but rather highlight the growing need to consider the presence and behaviors of pedestrians in the environment by designers, planners and decision makers (see, e.g., a recent report commissioned by the U.K. Cabinet Office on this subject [98]). In particular, public safety in crowded situations (e.g. concerts, religious or political gatherings) has become an important research area in the last years, with relevant contributions from physics, psychology, computer science and, of course, civil engineering. Acquiring data for this kind of study is obviously absolutely crucial for sake of understanding the implied phenomena and evaluating developed solutions for analysis, decision support, prediction. In video surveillance, scene modeling and understanding is also an important research area. Important tasks of scene modeling and understanding are (i) extracting motion information (e.g. trajectories), (ii) identification of entry and exit points of trajectories in the analyzed scene, (iii) characterization of the interaction of trajectories (highlighting, for instance, crossings or potential conflicts).

Pedestrians in videos taken from fixed cameras tend to appear and disappear at relatively precise and recurring locations, such as doors, gateways or particular portions of the edges of the scene. Moreover, pedestrian behavior in a given scene might imply waiting at a certain location then moving whenever certain conditions are met or given events happen. We refer to locations where pedestrians appear or start moving as *sources* (potential origins of a trajectory) and the locations where they disappear or stop moving as *sinks* (potential destinations). Traditionally, crowd analysis is performed by the analyst who manually identifies and detects different relevant activities in the scene. A portion of video is given to each analyst together with a list of events (behaviors) and objects to look for. The analyst informs the concerned authorities if any of the given events or objects are detected. Such kind of manual analysis of video is labor intensive, time consuming and prone to errors due to weak perceptive capabilities of humans, but also to the repetitiveness of the activity.

In this paper, we propose an approach for crowd behavior analysis (and, to a certain extent, understanding in the acceptation of the term adopted by [1]) adopting two novel algorithms, the first able to generate long, dense, reliable and accurate pedestrian trajectories and the second clustering them to generate long term reliable and abstract information describing flows in the whole video. The final results provide directly information characterizing flows but it also represents a starting point for further high-level analyses of crowd behavior. The approach starts by dividing the input video into multiple *segments* of equal length and, considering that the frame rate of the video is constant, duration. The initial frame of each segment is overlaid by a grid of particles initializing a dynamical system defined by optical flow, as discussed by [67]. Time integration of the dynamical system over a segment of the video provides particle trajectories (tracklets) that represent motion patterns in the scene for a certain time interval associated to the analyzed segment. We detect sources, sinks and main flows in the segment (for sake of brevity sometimes we will refer to this information as segment local *track*) by analyzing motion patterns followed by clusters of tracklets, obtained using an unsupervised hierarchical clustering algorithm, where the similarity is measured by the Longest Common Sub-sequence (LCS) metric. To achieve final global tracks, covering all the video, we cluster the achieved local tracks through the same hierarchical clustering algorithm. Our main contributions are: (1) Generating dense and long trajectories, (2) identifying sources and sinks, (3) understanding behavior of the crowd in the scene by considering full length video, (4) achieve the above results without requiring object detection, tracking, nor training, targeting employment in naturalistic conditions. The paper breaks down as follows: the following Section presents the current state of the art in the identification and characterization of pedestrian flows in crowded scenes, while Section 5.2 presents the overall proposed approach. Section 5.2.1 focuses on the algorithm to extract long, dense, accurate and reliable trajectories and Section 5.2.4 describes in details the clustering algorithm applied to generate local and global tracks. Section 6.4 describes the achieved experimental results, also by comparing the proposed approach with the most relevant existing alternatives. Conclusions and future developments end the paper.

5.2 Proposed Framework

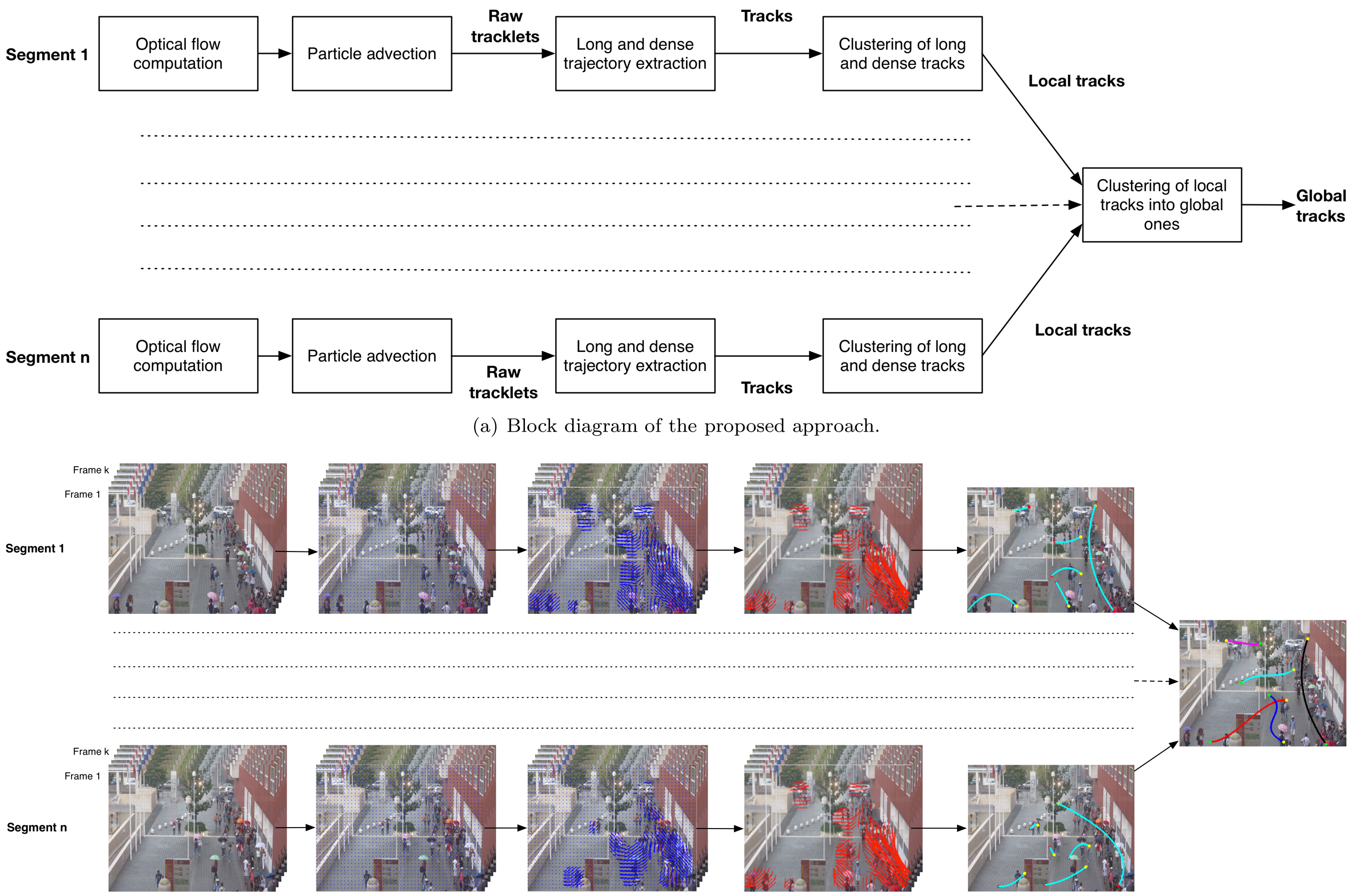
In this paper, we propose an approach for crowd behavior understanding adopting two novel algorithms, the first able to generate long, dense, reliable and accurate pedestrian trajectories and the second clustering them to generate long term reliable and abstract information describing flows in the whole video. The final results provide directly information characterizing flows but they also represent a starting point for further high-level analyses of crowd behavior. As shown in Figure 5.1, the approach starts by dividing the input video into multiple *segments* of equal length and duration, considering videos with a constant frame rate. The initial frame of each segment is overlaid by a grid of particles initializing a dynamical system defined by optical flow, as discussed by [67]. Time integration of the dynamical system over a segment of the video provides particle trajectories (*tracklets*) that represent motion patterns in the scene for a certain time interval associated to the analyzed segment. We detect sources, sinks and main flows in the segment (for sake of brevity sometimes we will refer to this information as segment *local track*) by analyzing motion patterns followed by clusters of tracklets, obtained using an unsupervised hierarchical clustering algorithm, where the similarity is measured by the Longest Common Sub-sequence (LCS) metric. Results achieved so far are intrinsically related to a single segment of the analyzed video, associated to a relatively short time frame. To achieve final *global tracks*, covering all the video, we cluster the achieved local tracks through the same hierarchical clustering algorithm. Our main contributions are:

1. generating dense and long trajectories,
2. identifying sources and sinks,
3. understanding behavior of the crowd in the scene by considering full length video,
4. achieve the above results without requiring object detection, tracking, nor training, targeting employment in naturalistic conditions.

5.2.1 Achieving Reliable Descriptive Motion Information

The input to our framework is a sequence of frames and, as summarized in the previous section, a first phase of the overall approach is aimed at achieving reliable descriptive motion information that will be then further processed to obtain local and global tracks. As already mentioned, we adopt an overall divide-and-conquer approach, splitting the overall frame sequence into n segments, each containing k frames. We then perform a segment local analysis to achieve tracklets that will be clustered later.

The first step to achieve tracklets is the computation of dense optical flow between two consecutive frame of every segment. We employ the method proposed by [99] where gray value constancy, gradient constancy, smoothness, and



(a) Block diagram of the proposed approach.

(b) Sample frames of the application of the approach to the case study.

Figure 5.1: The proposed framework for sources and sinks detection.

multi-scale constraints were used to compute highly accurate optical flow. Consider a feature point i in the frame associated to time t of a segment: its flow vector $Z_{i,t} = (X_{i,t}, V_{i,t})$ includes its location $X_{i,t} = (x_{i,t}, y_{i,t})$ and its velocity vector $V_{i,t} = (v_{x_i,t}, v_{y_{i,t}})$ (i.e. the velocity vector is made up of the change in the horizontal and vertical positions); moreover, for each feature point, we can compute θ_i, that is the angle or direction of V_i, where $0° \leq \theta \leq 360°$. Then $\{Z_1, Z_2, \ldots, Z_m\}$ is the motion flow field of all the foreground points of an image.

We can thus initialize a continuous dynamical system in which the velocity of a point at time t is essentially related to the optical flow of the same point, which is given by equation 5.1

$$V_{i,t} = F(X_{i,t}) \tag{5.1}$$

5.2.2 Particle Advection

The next step is to advect a grid of particles over the optical flow field, that corresponds to the time interval 1 to T for each segment. We launch a grid of particles over the first optical flow field of every segment and each initial position of the particle represents the source point. Ideally, the grid should have the same resolution of the frame and size of the particle is same as size of the pixel; nonetheless this would imply huge computational costs. To avoid this problem, we reduce the resolution of the grid by dividing it by a non negative constant: consider $res_x \times res_y$ the resolution of the image and $c > 1$; the resulting grid G will have a size $g_x \times g_y$ where $g_x = res_x/c$ and $g_y = res_y/c$.

Considering the initial location $X_{i,t} = (x_{i,t}, y_{i,t})$ of particles with $i \in G$, their next location $X_{i,t+1}$ at time $t+1$ can be computed by numerically solving the system of equations achieved by considering equation 5.1 for all the particles in G by using following approximation:

$$X_{(i,t+1)} = F(X_{(i,t)}) + X_{(i,t)} \tag{5.2}$$

To achieve a trajectory Ω_i for every particle $i \in G$, taking the form $\Omega_i = \{X_{i,1}, \ldots, X_{i,T}\}$, where T is the integration time, with $T = k$ (we will use time and frame number interchangeably), we need to compute a pair of flow maps ψ_x and ψ_y. These maps contain the initial position of each particle and all the subsequent positions computed according to the above equation, as discussed in [67].

The trajectory achieve by means of this process represents a movement from the initial position through time (and through frames) according to the optical flow. However, when this kind of analysis is carried out on an unstructured crowded scene (e.g. a subway corridor with pedestrians getting out and in a platform), where people move towards different and potentially changing directions, in many cases the particle trajectory could drift from a flow of pedestrians characterized by a certain direction to a spatially close but distinct and different flow, moving towards a significantly different direction. In this case, the trajectory is erroneous, since pedestrians do not actually change direction so quickly,

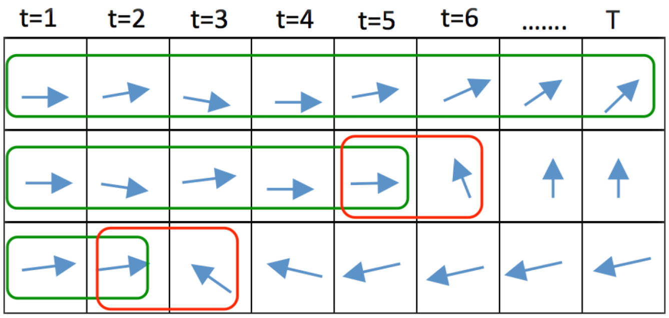


Figure 5.2: Flow associated to different particles: the first one is considered legitimate throughout the whole segment, whereas the second and third are trimmed due to significant changes in the flow direction in intermediate frames.

and this can effect the final outcome. Consider, for instance, Figure 5.2: each row shows the flow information at a given time for a given particle; the first row exemplifies a normal and legitimate trajectory, whereas the second and third rows show a situation that we consider an error, since the direction of the optical flow associated to the particle violently changes in too little time. The second and third particles, therefore, according to our approach will generate much shorter tracklets than those generated by current approaches (such as [67]) in which these changes are accepted.

More precisely, to avoid the above introduced defect, we modify equation 5.2 in the following way:

$$X_{(i,t+1)} = X_{(i,t)} + F(X_{(i,t)}) * B_i \tag{5.3}$$

$$B_i = \begin{cases} 1, & \text{if } \| \theta_{i,1} \text{ - } \theta_{i,t} \|_2 < \lambda \\ 0 & \text{otherwise} \end{cases}$$

The particle, therefore, continues moving forward if circular distance [100] between its initial direction $\theta_{i,1}$ computed initially and its direction at time t, $\theta_{i,t}$ is less than a specified threshold λ.

This approach, avoids errors due to particles drifting from a pedestrian flow to a different one, however the achieved trajectories are in general shorter in length than those extracted by [67]. In the previously mentioned Figure 5.2, the length of the extracted tracklets is equal to T frames only for the first particle whereas the other approach would always lead to tracklets of T frames: the number of frames of a tracklet is not necessarily an indicator of the actual length of the associated trajectory, but limiting the number of tracked frames inevitably leads to achieving shorter trajectories.

After particle advection, short duration particle trajectories called *tracklets* are obtained as shown in Figure 5.3. Some of these tracklets correspond to the background of the scene or noise and they are not actually part of our analysis. Therefore, in order to remove these tracklets, we estimate their actual length by

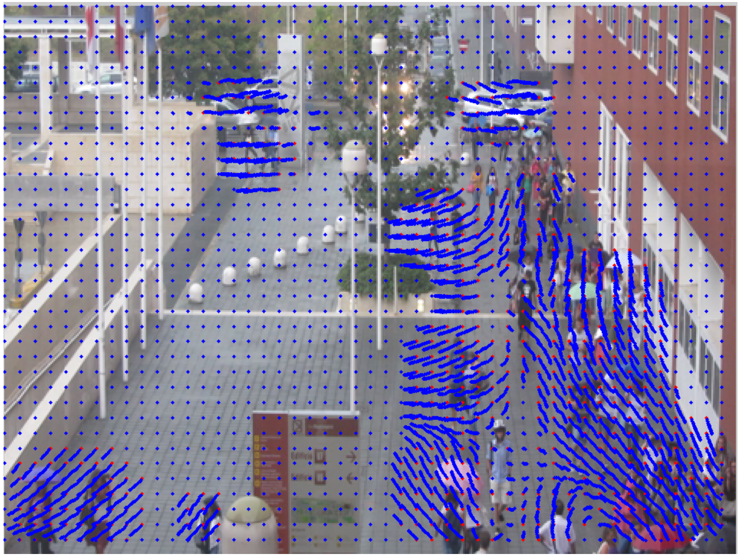


Figure 5.3: Tracklets achieved after particle advection.

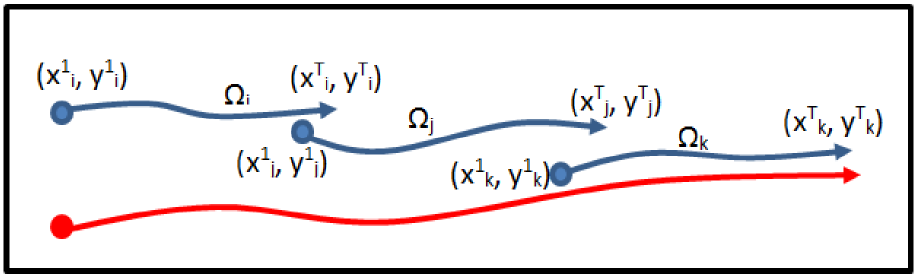


Figure 5.4: Example situation of generation of a long track from small tracklets.

computing the euclidean distance between the start and end points (remind that abrupt changes in direction of the particles block the trajectory construction, so most of the tracklets are very close to straight lines). We discard those tracklets for which $\| (x_i^1, y_i^1)$ - $(x_i^T, y_i^T) \|_2 < \delta$ (i.e. those tracklets whose length is very likely lower than a given threshold δ).

5.2.3 Clustering Tracklets to achieve Local/Global Tracks

Tracklets extracted through particle advection fail to represent important characteristics of the overall motion and they provide inadequate information for identifying source and sink points of the dominant flows and also for understanding behavior of the scene. To achieve these goals, we need dense and long trajectories which we can obtain thanks to the following assumptions: (i) a large number of tracklets corresponding to dominant flows is identified by the previous phase; (ii) source and sink points of tracklets associated to a common flow are often spatially close to each other. Our goal is to combine these accurate but generally short tracklets into longer trajectories. This induces a combinatorial matching problem that we define and solve *recursively* for all tracklets detected for each segment of the video sequence. The example frame shown in Figure 5.3 intuitively supports the claim that, for most scenes including a relatively large number of moving pedestrians, these two assumptions generally hold.

Let us now focus on the implications of the second assumption. Some of the achieved tracklets corresponding to single movements can be quite similar (in orientation), but their sources and sinks can be spatially different. Our goal is to combine similar tracklets into longer trajectories. For example, consider three tracklets Ω_i with source point (x_i^1, y_i^1) and sink point (x_i^T, y_i^T), Ω_j with source point (x_j^1, y_j^1) and sink point (x_j^T, y_j^T), and Ω_k with source point (x_k^1, y_k^1) and sink point (x_k^T, y_k^T) as shown in Figure 5.4. These tracklets start and end at different locations but the sink of one of them is spatially very close to the source of a different one: for instance, tracklet Ω_j starts very close to the sink point of Ω_i and the source of Ω_k is close to sink point of Ω_j. We exploit this spatial closeness of the tracklets in order to obtain longer trajectories. The similarity among the tracklets is computed by employing longest common sub-sequence algorithm, which will be discussed in details in the next section. The rationale of the approach, however, is that similar tracklets are identified and combined by means of k^{th} order least square polynomial regression as exemplified by the red line in Figure 5.4 (computed with $k = 3$). The outcome of the process, the red track, is therefore a long trajectory with source point (x_i^1, y_i^1) and sink point (x_k^T, y_k^T).

Let us now more formally define the above intuitive approach to achieve long tracks from shorter tracklets. First of all, we call a tracklet for which we would like to extract a longer trajectory a *query tracklet*. Let us now consider the analyzed scene: we have already overlaid a grid for particle advection, organizing the scene in "cells". The query tracklet will be positioned in a cell c, and we can define *neighbor tracklets* those ones positioned in the the Moore neighborhood of c. Finally, we call *candidate tracklets* those ones that we are considering for extending the query tracklet. The pseudo-code of the proposed algorithm is presented in Algorithm 1 and its description is reported here below.

The input to the algorithm is the *query tracklet* and output is the long trajectory; we also assume that the overall grid including the other tracklets is available as global information. The function *LongTrajectory* is divided into two steps: first of all, we collect all the tracklets that, due to to spatial arrangement, represent a plausibly connected path, but we also filter out tracklets that are not sufficiently similar to the starting one, because the resulting track would present an abrupt change of direction. This operation is executed by the *CompleteTrajectory* function that operates on an array L initialized by the calling environment (lines 2 and 3 of *LongTrajectory*) as containing only the query tracklet Ω_q. The function considers the tracklets present in the neighborhood of the query tracklet, and evaluates if they could represent a plausible continuation of the related path (line 4), inserting them into an array of candidate tracklets. If this array is not empty, the candidate tracklet that best matches the query one (line 9) is selected, inserted in the array L and then the *CompleteTrajectory* is recursively called considering the added query as next starting point (line 13). When the candidate tracklets array is empty the algorithm ends, returning the array L containing the tracklets that were added during the recursive execution. Finally, the resulting set of tracklets is then combined by means of k^{th} order least square polynomial regression (line 5 of *LongTrajectory*).

This function is applied to tracklets positioned in every cell of the grid. Some of them will basically not be extended at all; moreover some of the achieved tracks will be actually very similar to portions of larger ones: by definition of the algorithm, in fact, the existence a track spanning across $k > 2$ cells makes it very likely that additional $k - 2$ shorter (but not atomic) tracks are achieved later on, considering cells explored during the first computation.

The set of achieved tracks still contains also short tracklets, for which no extension was possible. The goal of this step, however, is to obtain dense and long trajectories covering all the scene and representing the most significant motion patterns, therefore we can filter out tracks that are shorter than a given threshold, analogously as we did to remove noise in the particle advection step (in this case the euclidean distance estimation of the actual length of the track is even more plausible since the considered tracks are, by construction, quite smooth).

Figure 5.5 shows the achieved long and dense tracks with increasing thresholds: the number of tracks decreases with growth of the threshold, but trajectories are still dense enough to represent whole motion of the scene even at the higher thresholds. Even though it is of course important to avoid setting a threshold so high that tracks representing important flows are filtered, it must be noted that the reduction of the number of tracks simplifies the computation associated to subsequent steps of the overall approach without suppressing important information.

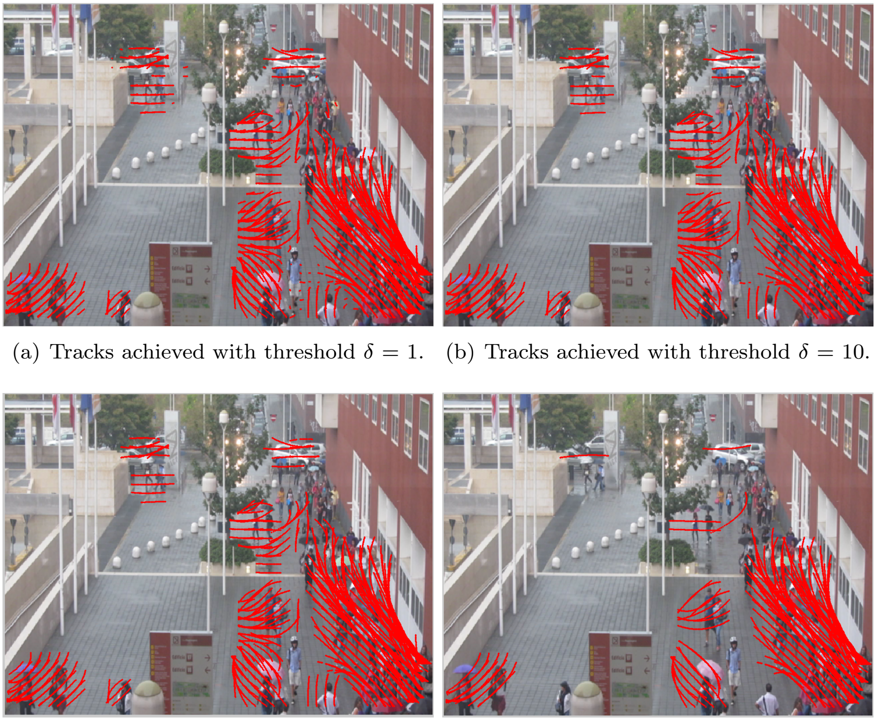
After achieving dense tracks, the next step is to combine similar tracks into local tracks by adopting novel hierarchical clustering algorithm. The classical supervised clustering algorithms can not be used as the number of flows (and therefore desired clusters) are unknown. Therefore, we propose a novel hierarchical clustering algorithm, based on the following procedure.

1. We sort the tracks in descending order on the basis of their length; in particular, let $T_L = \{t_1, t_2, \dots, t_k\}$ represent the sorted list of tracks and $\{l_1, l_2, \dots, l_k\}$ the respective length of tracks, we have than $l_i < l_j$ with $1 \leq i < j \leq k$.

2. We set up a list of tracks to be considered T_R, which initially is the complete list of tracks excluding first track t_1; we also set up a list of clusters L_C, initially containing first track t_1 (the longest one) that is also used as initial cluster center;

Algorithm 1 Generating long tracks starting from tracklets

```
Input: tracklet $\Omega_q$
Output: track $L_t$
function LongTrajectory($\Omega_q$)
    initialise array $L$ as empty
    insert $\Omega_q$ in first position of $L$
    $L$ = completeTrajectory($\Omega_q$,$L$)
    return polynomial regression on $L$
end function
Input: tracklet $\Omega_q$, array of tracklets $L$
Output: array of tracklets $L$
function CompleteTrajectory($\Omega_q$, $L$)
    initialise array $C$
    for all tracket $t$ in neighborhood of $\Omega_q$ do
        if $\|X_q^T - X_t^1\| \leq \epsilon$ then
            insert $t$ in $C$
        end if
    end for
    if $C$ is not empty then
        $bestCandidate = \arg\max_{c \in C} \frac{LCS(L[0],c)}{min(Len(L[0]),Len(c))}$
        $match = \frac{LCS(L[0],bestCandidate)}{min(Len(L[0]),Len(bestCandidate))}$
        if $match > m_t$ then
            insert $bestCandidate$ in tail of $L$
            return CompleteTrajectory($bestCandidate$,$L$)
        end if
    else
        return $L$
    end if
end function
```



(a) Tracks achieved with threshold $\delta = 1$.
(b) Tracks achieved with threshold $\delta = 10$.
(c) Tracks achieved with threshold $\delta = 20$.
(d) Tracks achieved with threshold $\delta = 50$.

Figure 5.5: Tracks achieved after the application of the algorithm to generate long tracks from shorter tracklets with increasing length thresholds.

3. We select the shortest track t_s from the list T_R, and compare it with cluster L_C using longest common sub-sequence metric, that will be described in the following Section, to compute similarity measure. If this value is greater than a threshold φ, then track t_s is assigned to the cluster and we delete the track t_s from the list T_R.

4. If the cluster's size exceeds a positive value of S we consider that sufficient information about the associated flow has already been achieved; therefore we identify source and sink location and update the center of the cluster by using K^{th} order least square polynomial regression. We used $S = 30$ in our experiments. The source and sink of cluster are selected according to a simple procedure: the selected pair is made up of the source point of a tracks and the sink point of (generally another) track that are part of the cluster and, in particular, the pair for which the euclidean distance between source and sink is maximum. The updated cluster L_C is assigned to list of global tracks T_G, which is initialized to be empty initially.

5. We repeat the previous step until T_R is not empty.

A pseudocode of the above clustering algorithm is described in Algorithm 2.

Algorithm 2 Clustering Local Tracks into Global Tracks

```
Input: list of local tracks $T_L$
Output: list of global tracks $T_G$
function ClusterTracks($T_L$)
    sort $T_L$ according to length in descending order
    cluster $L_C$ = first element of $T_L$
    list of remaining tracks $T_R = T_L$ - $L_C$
    $T_G = 0$
    repeat
        for all each track $t$ in $T_R$ do
            if matching ratio between $t$ and $L_C > \varphi$ then
                add $t$ to $L_C$
                if size of $L_C >$ S then
                    update cluster center for $L_C$
                end if
                remove $t$ from $T_R$
            end if
        end for
        update cluster center for $L_C$
        add $L_C$ to $T_G$
        $L_C$ = largest track $t$ in $T_R$          ▷ next largest track in $T_R$
        remove $t$ from $T_R$                       ▷ remove track assigned to $L_C$
    until $T_R$ is not empty
    add $L_C$ to $T_G$                              ▷ $T_G = 0$ in final step
end function
```

5.2.4 Longest Common SubSequence Computation

At this stage, we define similarity measure for comparing and clustering similar trajectories. There are number of approaches for measuring similarity of the moving object trajectories, such as [101] and [102]. A survey of different similarity measures for trajectory clustering is reported by [103]: Euclidean and Dynamic Time Warping (DTW) approaches are more sensitive to noise whereas Longest Common Sub-Sequence is efficient for series of unequal lengths and it is more robust to noise and outliers than DTW, as discussed by [104] and by [105].

The key idea of LCS is to match two time-series of tracklets by not considering all points of the tracklets, that can, to a certain extent, have different lengths. The following procedure allows verifying to which extent two trajectories can be considered similar (or *matching*, according to a certain similarity measure) and therefore what is the longest portion they have in common.

Let T_1 and T_2 represent two tracklets with size n and m respectively: $T_1 = \{(x_t, y_t), t = 1, ..., n\}$ and T_2 having analogous structure but m elements; with $T_1(i)$ we denote (x_i, y_i) with $0 \leq i \leq n$ and analogously for T_2. We compute the similarity among two tracklets by recursively finding a matching M between portions of these trajectories using a dynamic programming procedure that we will only briefly introduce for sake of space. Two constants are needed, respectively Φ controlling matching sequences in time, determine as $\Phi = \frac{\max(Length(T_1), Length(T_2))}{2}$ and Ω which is the spatial matching threshold. Formally the matching matrix M comparing T_1 and T_2 can be computed recursively as follows:

$$M_{i,j} = \begin{cases} 0, & \text{if } i \text{ or } j \text{ are } 0 \\ 1 + M_{i-1,j-1}, & \text{if } \parallel T_1(i) \text{ - } T_2(j) \parallel_2 < \Omega \text{ and } \mid i - j \mid < \Phi \\ max(M_{i-1,j}, M_{i,j-1}), & \text{otherwise} \end{cases}$$

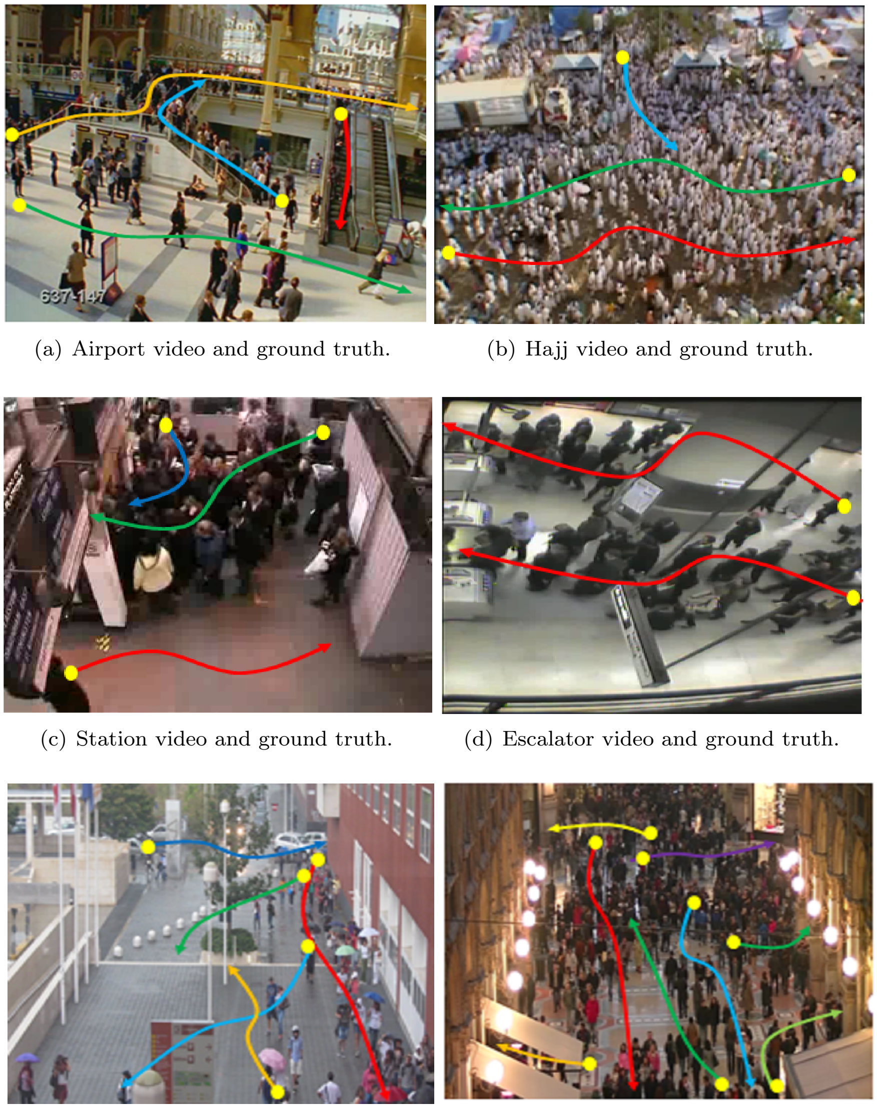
The similarity measure between two tracklets T_1 and T_2 is therefore $S(T_1, T_2) = \frac{LCS(T_1,T_2)}{min(n,m)}$, where LCS is the number of matching points between T_1 and T_2, according to the above matching matrix.

5.3 Experimental Results

This section presents qualitative and quantitative analyses of the results obtained from experiments on the application of the proposed approach to video sequences made available from other research groups and acquired through field observations. In particular, we carried out our experiments on a PC of 2.6 GHz (Core i5) with 4.0 GB memory, running a Matlab implementation of the presented algorithms; the analyzed data set includes videos made available from other research groups and described in [106, 70], in addition to videos we acquired in past researches described in [107, 108]. The overall set of video includes situations including both the so called structured and unstructured crowds [42](i.e. situations with respectively stable and varying flows in the scene), and very different density conditions.

The analyzed videos we will discuss in the remainder of the section are the following:

- *airport* video, Figure 5.6(a) [106]: this sequence shows a portion of an airport, including stairs and escalators, with relatively stable pedestrian flows in medium-low density conditions;
- *Hajj* video, Figure 5.6(b) [106]: this sequence was taken in the context of the yearly pilgrimage to Makkah, Saudi Arabia, and it shows a very high density situation in which the overall velocity of pedestrians is very low but characterized by three main and relatively stable movement directions;
- *station* video, Figure 5.6(c) [70]: this footage shows a platform in which pedestrians try to get on and off of a train; flows change in time due to



(a) Airport video and ground truth.

(b) Hajj video and ground truth.

(c) Station video and ground truth.

(d) Escalator video and ground truth.

(e) University video and ground truth.

(f) Gallery video and ground truth.

Figure 5.6: Dataset of analysed videos and associated manually defined ground truth.

the congestion that arises near one of the entrances of the wagon, and the density conditions are very different in distinct areas of the scene;

- *escalator* video, Figure 5.6(d) [70]: this is a footage of a portion of a platform in which two main flows lead to and from an escalator; the density conditions are medium-low and the flows are quite stable, although occlusions due to the presence of a column and other infrastructural elements are present in the scene;

- *university* video, Figure 5.6(e) [107]: this sequence shows the arrival of students that are going to undertake an admission test to a bachelor course at the University of Milano-Bicocca; the density conditions are medium-low but the number of pixels per person is quite low and many occlusions are possible also due to the presence of infrastructural elements;

- *gallery* video, Figure 5.6(f) [108]: this footage was taken in a commercial/turistic gallery in Milan's city center, in a Saturday afternoon; the density conditions are medium-high and the point of vantage causes a very high number of occlusions, also due to the irregular and varying nature of pedestrian flows.

All of the above figures also report a manual annotation describing the dominant flows identified by a human observer, that can be qualitatively compared to the achieved results, that will be presented later on. Different color codes are used for representing different flows while source points are always marked with yellow circles.

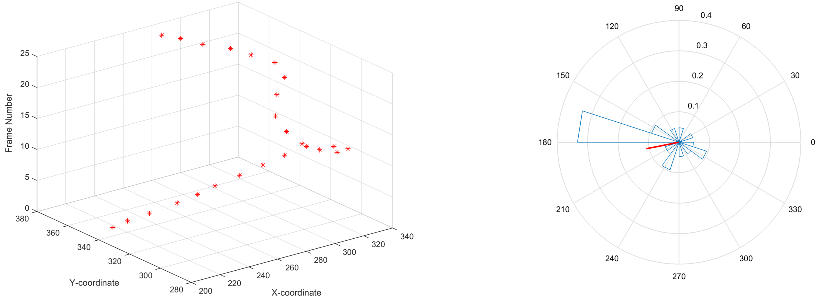
Since our framework consists of two major parts, the first aimed at generating dense long trajectories from short and accurate ones, the second able to detect sources and sinks of dominant flows, we describe two types of experiments. In the following subsection, we compare our method for the extraction of long and dense trajectories with baseline tracking techniques, in particular we consider KLT and SIFT based trajectories by analyzing all of the above mentioned videos. In this case, we adopt both a qualitative and quantitative approach, by showing the generated trajectories and also by comparing the number of trajectories extracted employing different thresholds for their length, to evaluate the capability of the approach to generate sufficiently long trajectories.

In subsection 5.3.2, instead, we describe the overall results about the detection of sources and sinks of dominant flows and we discuss them considering results achieved in those situations by state-of-the-art methods.

5.3.1 Comparison with Baseline Trajectories Extraction Approaches

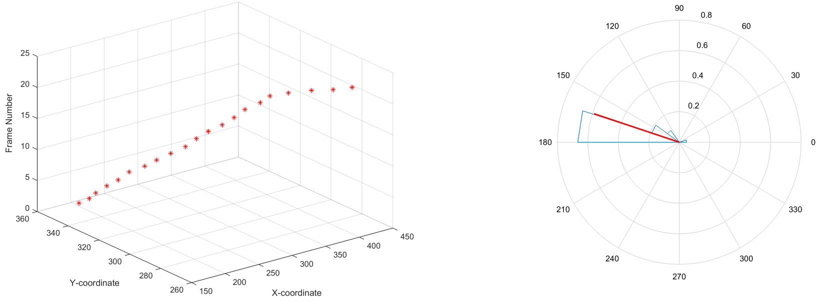
In order to evaluate improvement obtained with our proposed framework, we compare our method of generating long and dense trajectories with state of the art trackers considered as a baseline: in particular, we consider KLT trajectories adopted by [70, 71, 59, 61], SIFT trajectories adopted by [61, 42], approach described by Solmaz et al. in [67].

Due to the unavailability of consolidated ground-truth data for this kind of application, it is difficult to evaluate and compare the precision and, more generally speaking, *quality* of the results achieved by the proposed approach and baseline trackers. We propose here a combination of different quantitative and



(a) Temporal Plot of Trajectory. (b) Histogram of Trajectory.

Figure 5.7: Trajectory in Error.



(a) Temporal Plot of Trajectory. (b) Histogram of Trajectory.

Figure 5.8: Stable Trajectory.

qualitative measurements both in the above mentioned videos and in additional situations.

Quantitative Analysis

In particular, we first examine the performance of the above approaches when analyzing a synthetic rendered video. High quality rendered videos should incorporate deforming objects, complex light reflectance, camera motion, optical artifacts which make mimicking the real world videos very hard and challenging. However, our intention with this test is to evaluate the ability of the above approaches to consider some background considerations and knowledge about pedestrian movement employing an extremely simple video including a particle following different trajectories, to isolate the conceptual analysis the related paths from technical issues of trackers. In fact, trajectories extracted from complex videos in crowded environments imply errors and noise due to the severe occlusions and we want to be able to filter out erroneous paths.

Trajectories belonging to one motion pattern (e.g. the trajectory of the head of a pedestrian) may drift and become the part of different motion patterns (e.g. the trajectory of a body part of another pedestrian moving in a different direction). We call these trajectories as erroneous or occluded. The effect of this kind of occlusion is schematized in Figure 5.7: in particular, Figure 5.7(a) plots a trajectory extracted from a 25 frames synthetic video of a simple particle moving in the captured area, while Figure 5.7(b) shows the orientation histogram of the trajectory. Since the frame rate of the video is 25 frames per second, this trajectory is associated to just one second and therefore, considering normal human locomotion, it should not present a wide variety of orientations, but rather a main direction with relatively little changes. The orientation histogram in Figure 5.7(b) instead reports a wide range of orientations, highlighting the fact that the trajectory either belongs to noise or occluded with different motion patterns. In contrast, a more stable trajectory is shown in Figure 5.8, and it is characterized by a limited range of orientations as shown in Figure 5.8(b). On the basis of these considerations, we defined a plausibility test for each individual trajectory extracted from the proposed approach and other baseline methods. For computing this plausibility factor for a given trajectory $T\{X, \theta\}$, where X represents the spatial locations and θ represents orientations of the trajectory, with T containing k points, we perform following steps

1. Compute circular mean, i.e, θ_μ of θ as in [109] for the given trajectory T.
2. Compute circular distance from the mean for all trajectory points, i.e, $\mathit{CircDist}_i = (\theta_\mu - \theta_i)$, with $0 \leq i < k$ and where θ_i is i^{th} point of the trajectory.
3. Compute an indicator of smoothness

$$\mathit{Smooth}_i = \begin{cases} 1, & \text{if (} \theta_\mu \text{ - } \theta_i \text{) } < \Psi \text{ where } \Psi \text{ is set to 0.7854} \\ 0 & \text{otherwise} \end{cases}$$

4. Compute the overall trajectory smoothness indicator

$$\mathit{Smooth} = \frac{\sum_{i=0}^{k-1} \mathit{Smooth}_i}{k}$$

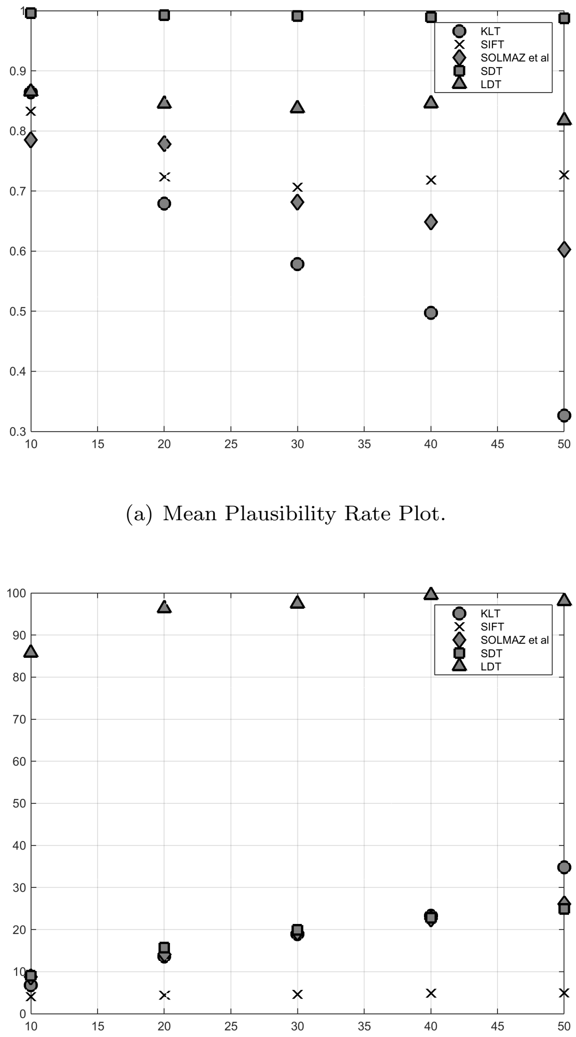
We consider a trajectory T as accurate if its smoothness indicator $\mathit{Smooth} \geq \gamma$, where γ is set to 0.5 in all the following experiments. This kind of test, for instance, would label as accurate the trajectory in Figure 5.8 but not the one in Figure 5.7.

By following this procedure we performed plausibility tests on all other trajectories extracted from the real world analyzed videos. We then compute an overall plausibility rate by computing the ratio between the number of accurate trajectories and the total number of extracted trajectories. The mean plausibility rate, mean length, minimum and maximum lengths of extracted trajectories for all methods using all the analyzed videos is summarized in Table 5.3.1. Results show that the sparse methods, i.e, KLT and $SIFT$, produce a relatively low number of trajectories compared to other particle based dense methods. KLT trajectories have approximately the same mean length as the dense methods but the plausibility is relatively lower. In case of $SIFT$, the mean precision rate is high but the mean length of the trajectories is too small as compared to other methods, and therefore, the trajectories extracted by this method could not be able to capture the whole motion of the scene. We also run Solmaz et. al [67] and our algorithm with the same configuration and by initializing the same number of particles for all the analyzed videos. The plausibility rate of trajectories extracted by our Short Dense Trajectories (SDT) method discussed in section 5.2.1 is very high relative to other methods but with approximately the same mean length, that would also be insufficient to describe the whole motion in the scene. We improve the mean length of trajectories by employing our Long Dense Trajectories (LDT) method discussed in section 5.2.3 by paying a small cost in terms of plausibility. In fact, plausibility rate for LDT is reduced because the tracks are clustered based on the similarity measure 5.2.4, which implies the potential connection of tracklets leading to a change in the flow direction and therefore to a less smooth but still plausible trajectory.

We further investigate the variation of performance of our and baseline methods with a changing segment size of the analyzed videos. We divide each analyzed video into five segments of different size, ranging from 10 to 50 frames. For each segment, we extract features (in case of $SIFT$ and KLT) or initialize particles (in case of other methods) in the first frame and tracked through last frame. In case of LDT, we extract tracks for each segment and then apply algorithm 2. The results of this analysis on the mean plausibility rate are shown in Figure 5.9(a): we observe that it generally drops with the growth of the segment size for all approaches but DLT whose plausibility decreases only slightly. As we already discussed, the plausibility rate of DLT is lower then DST but still higher than other methods. The mean length plot for the same analysis is shown in Figure 5.9(b): we observe that mean length slightly increases with the growth of segment size, but for DLT it remains almost constant. This means that this method is able to capture global motion information in the scene also with relatively small segments.

Table 5.1: Summary of mean plausibility and mean length of different methods

Methods	# of traj.	plausib.	mean length	max_length	min_length
KLT	2576	0.4973	23.2686	73.0258	2.0142
SIFT	3636	0.7268	4.9273	58.7974	2.0031
Solmaz et al	7633	0.6027	26.0615	86.8846	2.0104
SDT	7633	0.9876	24.9281	87.0447	2.0075
LDT	7633	0.8173	98.0238	320.8929	2.2279



(a) Mean Plausibility Rate Plot.

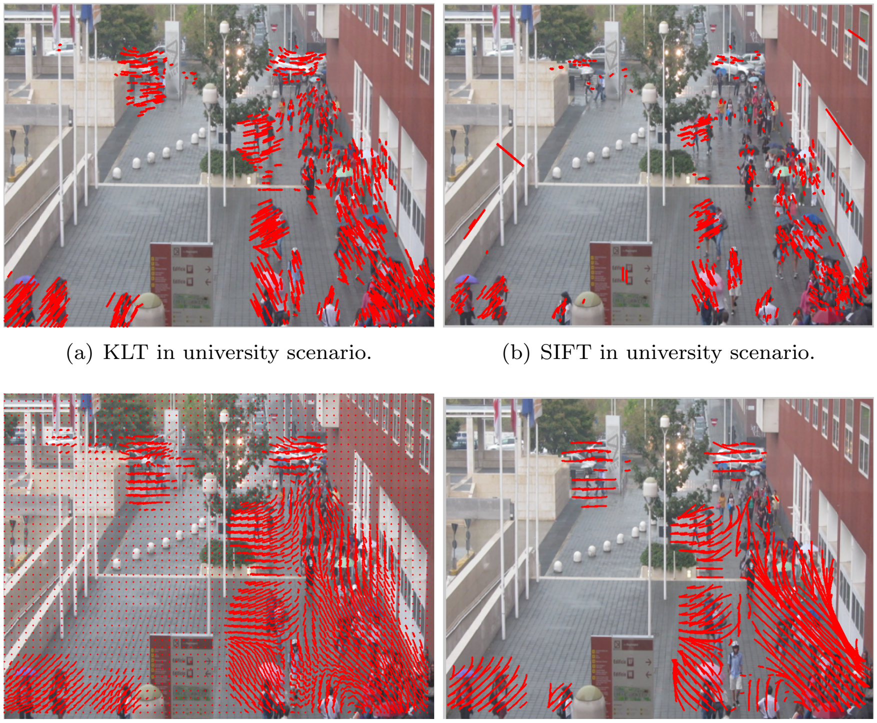
(b) Mean Length Plot.

Figure 5.9: Evaluation of mean plausibility and length of trajectories with different segment size.

Qualitative Analysis

The qualitative analyses will translate into understandable examples the implications of the above quantitative analysis.

In order to obtain KLT trajectories, we first identify low-level features (cor-



(a) KLT in university scenario.

(b) SIFT in university scenario.

(c) Solmaz et al. in university scenario.

(d) Proposed approach in university scenario.

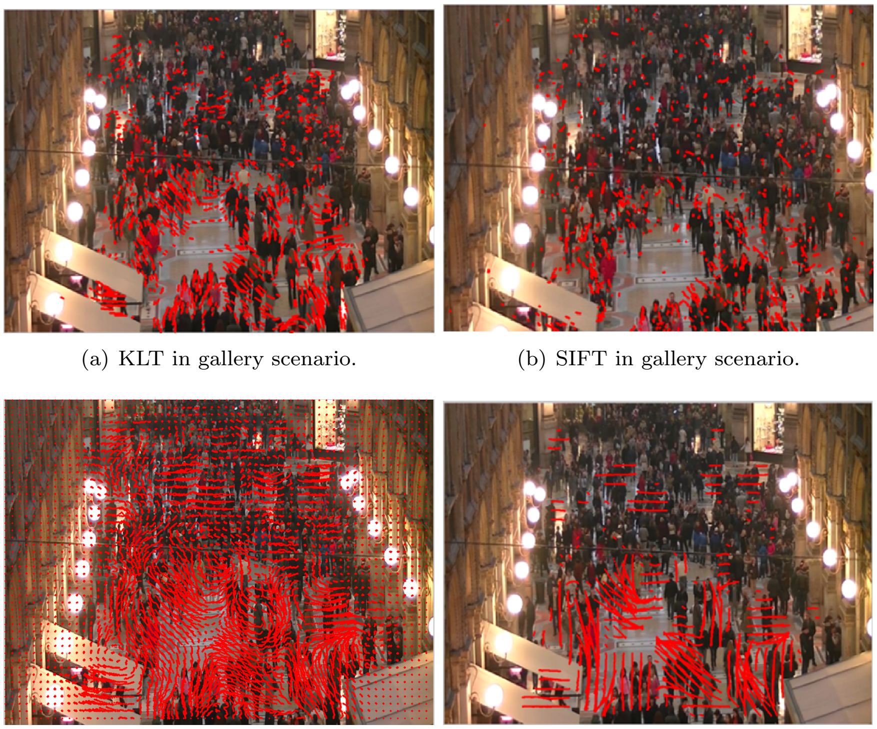
Figure 5.10: Comparison between state of the art trackers (KLT and SIFT) and the proposed approach in the university scenario.

ner points) in the initial frame using standard Shi-Tomasi-Kanade detector [110]. These corner points are tracked over time by using [93].

On the other hand, in order to extract SIFT trajectories, we first extract SIFT interest points from the initial frame; these points are then tracked through multiple frames by matching euclidean distance between SIFT descriptors within a neighborhood. More details about SIFT feature tracking can be found in [111].

Finally, trajectories generated by the application of [67] are more aimed at supporting crowd behavior understanding rather than implementing a traditional tracking system; due to this perspective, they represent the closest approach to the one we are proposing.

Before providing a quantitative analysis of the performances of the above approaches, a qualitative comparison in the university and gallery videos is provided in Figures 5.10 and 5.11: in the university scenario, SIFT is actually unable to generate trajectories in good accordance with the ground truth, and it generates even noticeable false positives, whereas KLT is able to generate plausible but short trajectories, due to the fact that features that are used for



(a) KLT in gallery scenario.

(b) SIFT in gallery scenario.

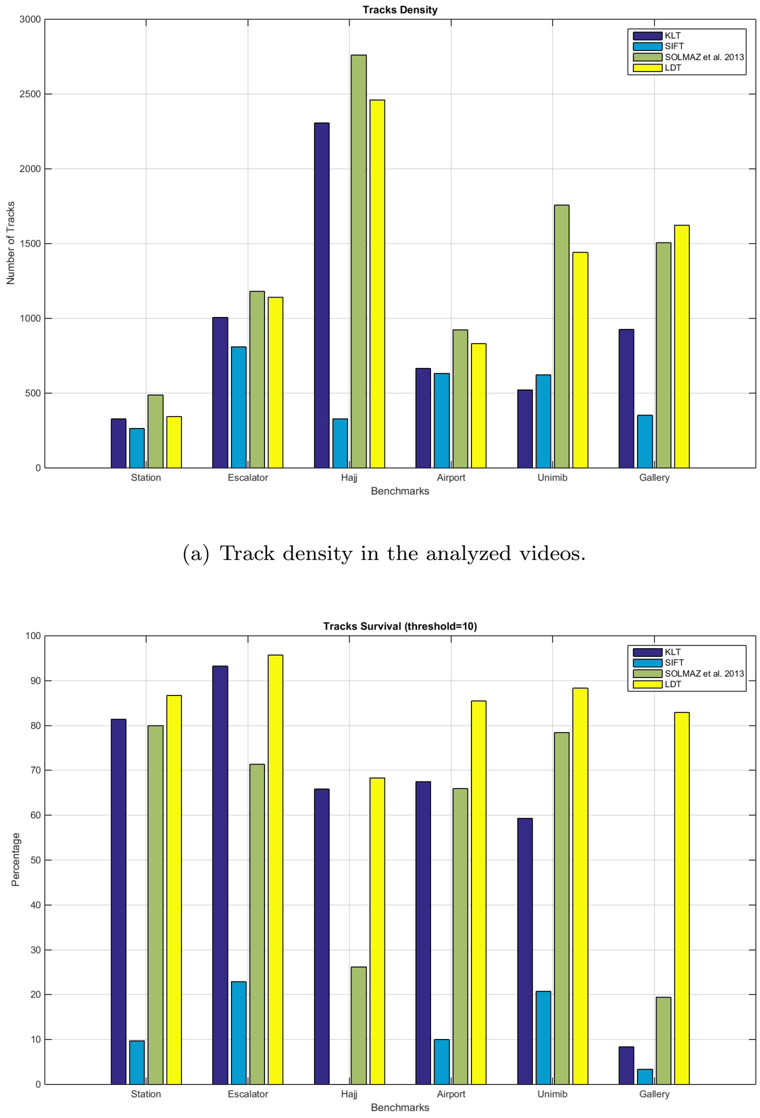
(c) Solmaz et al. in gallery scenario.

(d) Proposed approach in gallery scenario.

Figure 5.11: Comparison between state of the art trackers (KLT and SIFT) and the proposed approach in the gallery scenario.

tracking are not visible in every frame. The approach of Solmaz et al. [67], instead, produces results that are relatively similar to the trajectories generated by the proposed approach, although the trajectories are generally shorter and sometimes erroneous (i.e. continuous but associated to paths that are not really associated to real pedestrian flows). This difference is due to additional rules in our approach that avoid the generation of long tracks when base tracklets have different orientation, and it is even more apparent in the gallery scenario. In this situation both SIFT and KLT fail, since this video is extremely problematic for feature–based approaches due to dynamic occlusions and clutter, whereas Solmaz et al. [67] produces an extremely high number of tracks basically due to the fact that optical flow in the walkable area of the gallery is dense and in a large number of varying directions. The additional rules for filtering non plausible trajectories we included in the proposed approach are instead able to consistently reduce this noise.

As above two trackers produce trajectories based on feature points extracted from the initial frame of video segment, therefore these trajectories are sparse. Another problem with feature base trajectories is that in high density situations,



(a) Track density in the analyzed videos.

(b) Track survival with threshold 10 in the analyzed videos.

Figure 5.12: Quantitative analyses about track density and survival comparing the proposed approach and state-of-the-art tracking approaches.

due to complex movement of people, and due to severe inter and intra object occlusions, feature points can not be tracked for long period of time. Therefore, in high density situations, feature base trajectories are short. These short and sparse feature based trajectories are inadequate to capture crowd dynamics.

For a quantitative comparison of the results of these approaches, we report in Figure 5.12(a) a graphs describing the track density in different scenarios, that is, the raw number of tracks generated by the different approaches after removing noise and tracklets whose length is less than 2. Per se, this metric is not actually an indicator of success of the approach, nonetheless the very low number of tracks generated by the SIFT approach is an indicator that it is simply unable to grasp the fundamental motion information of the scene. In Figure 5.12(b), instead, we show the percentage of the above tracks that are longer that a threshold set to 10: once again, SIFT is not adequate to this task since even if the produces tracks are few, most of the produced ones are not even long. The other approaches perform similarly in most of the scenarios, in case of medium-low density and/or structured crowds (i.e. with flows of pedestrians that are neatly separated and generally stable), whereas some difference can be highlighted in the university and gallery videos. In these cases, flows of pedestrians actually mix and cause occlusions (generating problems to KLT) and a very high number of possible ways of connecting optical flow tracklets (for Solmaz et al. [67]).

To further characterize these differences, we also report in two extremely different scenarios, station and gallery, the variation in the survival rate of generated tracks with the growth of the length threshold. Results of this analysis are shown in Figure 5.13: while for the station video Solmaz et al. [67] produce a percentage of surviving tracks that is very similar to our approach, and quite higher than both SIFT and KLT, in the gallery scenario the difference between the survival tracks ratio is already significant for a length threshold set to 10.

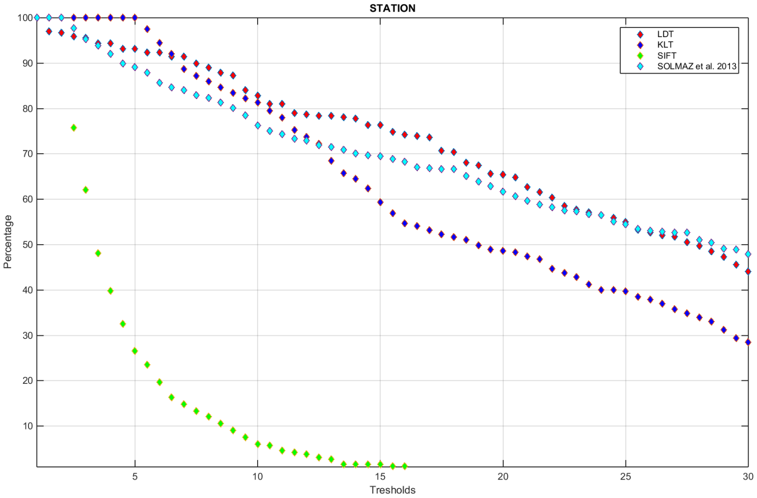
5.3.2 Identification of Sources and Sinks, and Characterization of Dominant Flows

In this section, we present results of our proposed framework with reference to the capability to identify sources, sinks (as defined in the introduction) and in general to characterize pedestrian flows in the scene. We analyze different videos to highlight different features and discuss the performance of the proposed approach also with reference to current approaches to this problem present in the literature. Once again, we propose both quantitative and qualitative analyses.

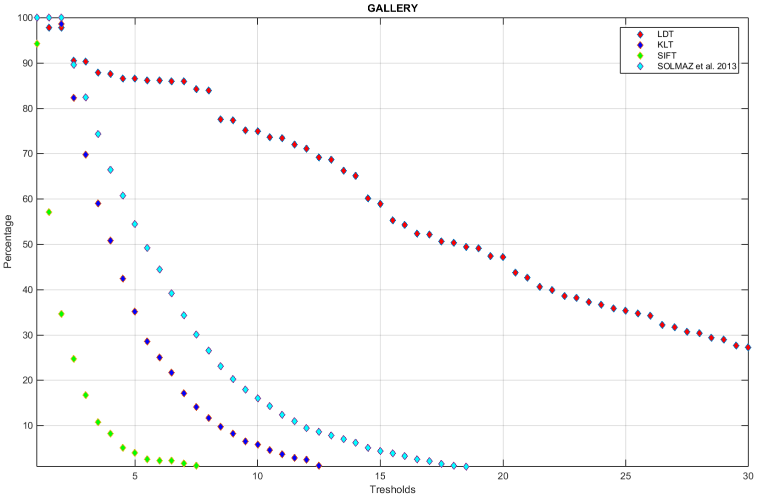
Quantitative Analysis

The input to our framework is represented as a sequence of frames and we divide each video sequence into different temporal segments. The length of each analyzed video is 350 frames and we set the length K of each segment equal to 50 frames.

We extract global flows by using the trajectories extracted by using our approach and other baseline methods and finally applying our clustering algorithm 2. In order to quantify the accuracy of each method after employing our clustering algorithm, we compare the achieved results with ground truth global



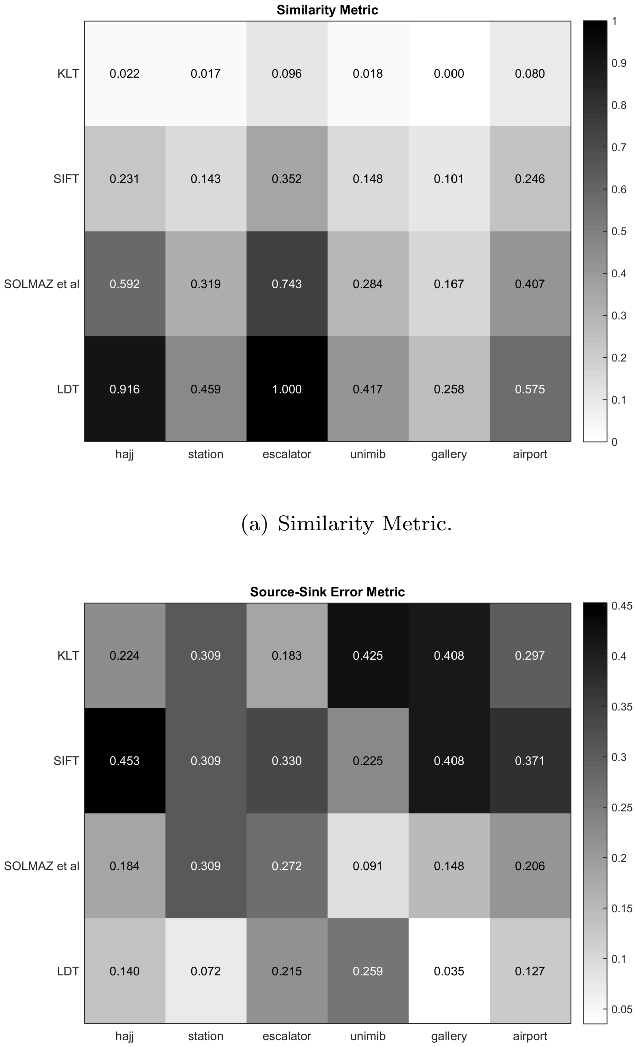
(a) Percentage of remaining tracks with growing thresholds in the station video.



(b) Percentage of remaining tracks with growing thresholds in the gallery video.

Figure 5.13: Quantitative analyses comparing the survival of trajectories with varying length threshold in station and gallery videos for the proposed approach and state-of-the-art tracking approaches.

flows. We obtained ground truth data for each analyzed video by manual identification of global flows: the visual plot of manually detected global flows for



(a) Similarity Metric.

(b) Source/Sink Locations Error Metric.

Figure 5.14: Similarity and source/sink error metrics comparison between proposed approach and baseline methods.

each analyzed video is shown in Figure 5.6.

Since to the best of our knowledge there is no agreed upon mechanism for evaluating the accuracy in the detection of sources and sinks, and in the characterization of main flows in a scene, we introduced and computed two metrics and in particular: (1) flow *similarity* metric and (2) source/sink error metric.

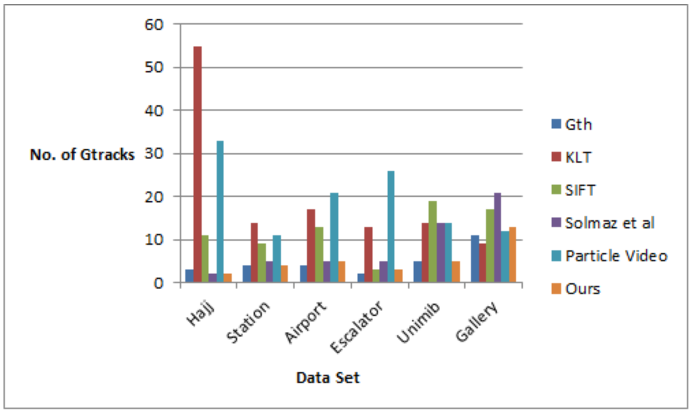


Figure 5.15: Total number of global tracks found for each method by our clustering algorithm.

We define and compute flow similarity metric by comparing global flows *Gtrack* detected by each tracks generation method followed by our clustering algorithm with ground truth *Gth*. The similarity is measured exploiting LCS and by using the following equation:

$$Sim = \frac{\left(\sum_{i=1}^{N} \arg\max_{j\in[1,M]} \frac{LCS(Gtrack_j, Gth_i)}{Length(Gth_i)}\right)}{N} \tag{5.4}$$

In particular, N represents total number of ground truth tracks while M represents total number of global tracks detected by method for the analyzed video. The equation considers all the actual N global tracks inn the ground truth data and selects the extracted track that is most similar to the ground truth.

We observe that, in this experimentation, $N \leq M$ uniformly for all approaches; this is likely caused by the fact that our clustering approach works very well with long and dense tracklets but cannot merge into a single cluster tracklets that are too short and not similar according to LCS. We also observe this kind of situation in clustering tracks achieved with baseline methods, since these methods generally produce small and implausible tracks in contrast to DLT that is generally able to capture each dominant motion and to produce almost the same number of global tracks present in the ground truth as shown in Figure 5.15.

We computed mean similarity *Sim* for each analyzed video adopting the different track generation techniques and results are shown in Figure 5.14(a), which supports both a quantitative and qualitative evaluation: darker blocks, in fact, show that global tracks identified by our proposed method is closer to the ground truth than the lighter blocks, associated to the baseline techniques.

The second metric simply measures how far the source/sink locations of extracted global tracks from the source/sink locations of ground truth data. The simplest way to compute this metric would be to calculate euclidean distance between the source/sink locations of global tracks and source/sink location of ground truth tracks for the analysed video. However, this is implausible for mostly two reasons: first of all, distance expressed in pixels is dependent on the type of scene and not necessarily proportional to actual errors in the real world, due to perspective; second, it is very hard to provide a way to normalize in a sensible way this kind of metric.

Therefore, in order to alleviate this problem, we build an *Association Matrix* that captures the joint probability distribution of source and sink locations of all the trajectories in the analyzed video. Actually, we build two matrixes, one constructed employing the ground truth data and another employing the global trajectories extracted automatically that present the best match to ground truth ones, as for the similarity metric.

In order to build this matrix, we assume two discrete jointly distributed random variables **X**, representing “source” locations of the trajectories and **Y** representing “sink” locations. An *Association Matrix* for n trajectories is shown below.

$$P(X,Y) = \begin{Bmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1n} \\ p_{21} & p_{22} & p_{23} & \cdots & p_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & p_{n3} & \cdots & p_{nn} \end{Bmatrix}$$

Each row/column of an *Association Matrix* shows the probability distribution of the source and sink points of single trajectory P_k over all other n trajectories in the analyzed video. Let P_k is the distribution of sources and sinks of trajectory k with all other n trajectories and represented as $\{p_{k1}, p_{k2}, p_{k3}, \ldots, p_{kn}\}$, where $p(k,j)$, is the joint probability of start and stop locations for trajectories k and any trajectory j. We use a Gaussian likelihood model [112] to compute the probability of a trajectory k to start (or stop) from the initial (or final) location of a (potentially different) trajectory j in the scene as equation 6.1:

$$p_x(k,j) = \mathrm{e}^{-\| \frac{x_k - x_j}{\sigma} \|} \tag{5.5}$$

Where x_k and x_j are the source (or sink) locations respectively of trajectory k and j. Assuming independence among the trajectories, we multiply the above values for start and sink points to calculate joint probability $p(k,j)$ for trajectories k and j. In the same way, we compute joint probabilities of all other trajectories and after normalization, we obtain an *Association Matrix*.

Following this procedure, we computed *Association Matrixes* for ground truth tracks (AM_{Gth}) and the selected global tracks produced by the compared methods (AM_{Gtrack}), in all the analyzed videos. Finally, we computed the difference between the *Association Matrices* by using *Kullback-Leibler (KL) divergence*, also known as relative entropy, denoted by $D_{KL}(AM_{Gth}||AM_{Gtrack})$, computed by using equation 6.4. The value $D_{KL}(AM_{Gth}||AM_{Gtrack})$ is associated to the loss of information caused by using *Gtrack* instead of ground truth data *Gtrack*, and it should be considered, therefore, an indicator of how distant the results are from the ground truth.

$$D_{KL}(AM_{Gth}||AM_{Gtrack}) = \sum_i AM_{Gth}(i) \ln \frac{AM_{Gth}(i)}{AM_{Gtrack}(i)} \tag{5.6}$$

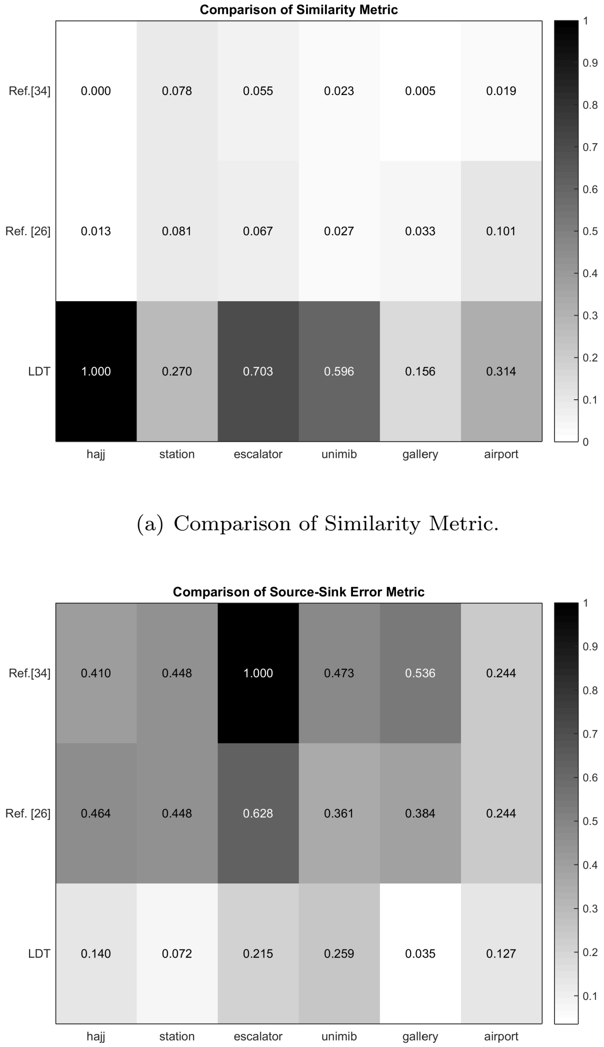
Figure 5.14(b) reports the values of the above metric for evaluating the distance between the source/sink locations achieved with the proposed method and other baseline approaches from ground truth. This metric is associated to an error, so the low values indicate that source/sink of the global tracks lie close to the ground truth. Results are in line with those related to the flow similarity metric.

Finally, we also compare our method with most relevant state of the art techniques, i.e, [105] and [70], in a quantitative way. Both these methods use KLT method for extracting trajectories from the crowded scene followed by clustering algorithm. Since KLT is a sparse method the extracted trajectories are unreliable and short enough to cover just essential parts of the motion in the scene. Another limitation with [105] is that during clustering phase, instead of updating the cluster center, the long trajectory among the set of clustered trajectories is selected as new cluster center. In this way, several trajectories representing the single true flow appeared at the end of clustering. The clustering algorithm is improved in [70] by updating the cluster center, however, the trajectories produced at the end of clustering are still short and they appear as different parts of a single unique actual flow. As shown in 5.16(a), methods [105] and [70], produce very low similarity values. The reason is that the trajectories produced by these methods are short and hence equation 5.4 gives very low values, and [70] does not provide significant improvement over [105].

Qualitative Analysis

Local tracks can be considered as a by-product of the overall process, but they can represent useful indications of changes in the situation between different time slices associated to the different segments. For instance, Figure 5.17 shows different local tracks associated to different segments of the station video as well as the overall global detected tracks: the overall flows are detected correctly (qualitatively comparing Figure 5.18(c) with Figure 5.6(c)), moreover during the analysis, some flows detected in a given segment (i.e. Figure 5.17) are not detected in a following one. This kind of event, beyond the specific situation, could be a signal that could be interpreted by a higher-level module, performing semantic analysis of the results, indicating that an area is changing from free flow to a congested state.

Other situations, similarly characterized by medium-low density situations and relatively stable flows, yield similar results: Figure 5.18(a) shows that in the airport video the main flows are correctly detected in a multi-floor scenario; some of them are actually correlated, as one merges into another: pedestrians climbing two staircases actually merge into a single flow in a T-junction, but they are



(a) Comparison of Similarity Metric.

(b) Comparison of Source/Sink Locations Error Metric.

Figure 5.16: Comparison with state of the art source and sink identification methods.

detected as two flows. In an analogous way, the university video is also correctly analyzed, in terms of detection of main flows, as shown in Figure 5.18(e), but in this case one of the detected flows is actually generated as a split from another larger one. These considerations also call for a subsequent phase of semantic analysis after the algorithm, in addition to a quantitative characterization of

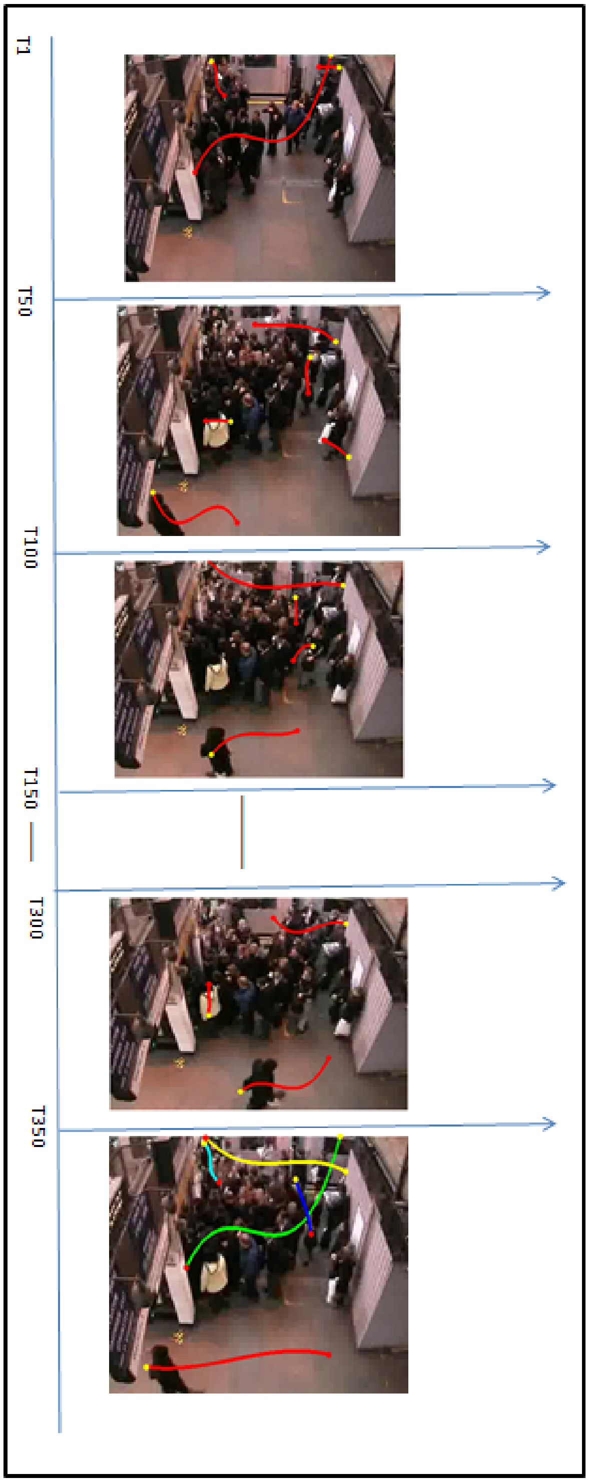


Figure 5.17: Local tracks resulting from analysis of segments and final global tracks in the station scenario.

the flows that would be necessary to actually define an O/D matrix.

The university video analysis also shows the fact that the proposed approach, if properly configured (i.e. with thresholds' values adequate to the specific scenario that is being analyzed), is robust to occlusions due to infrastructural elements that interrupt the visibility of a given flow of pedestrians: poles and tree branches, in fact, do not avoid the completion of tracklets into proper global trajectories. The escalator video analysis, though, shows that this robustness has limits: overall flows are in fact detected but the large obstacle (i.e. a column) combined with the value of the thresholds cause the upper flow to be split into two motion patterns. Adjusting the thresholds, in this case, could solve the problem but there could be downsides, such as the acceptance of incorrect trajectories representing implausible completions of short tracklets.

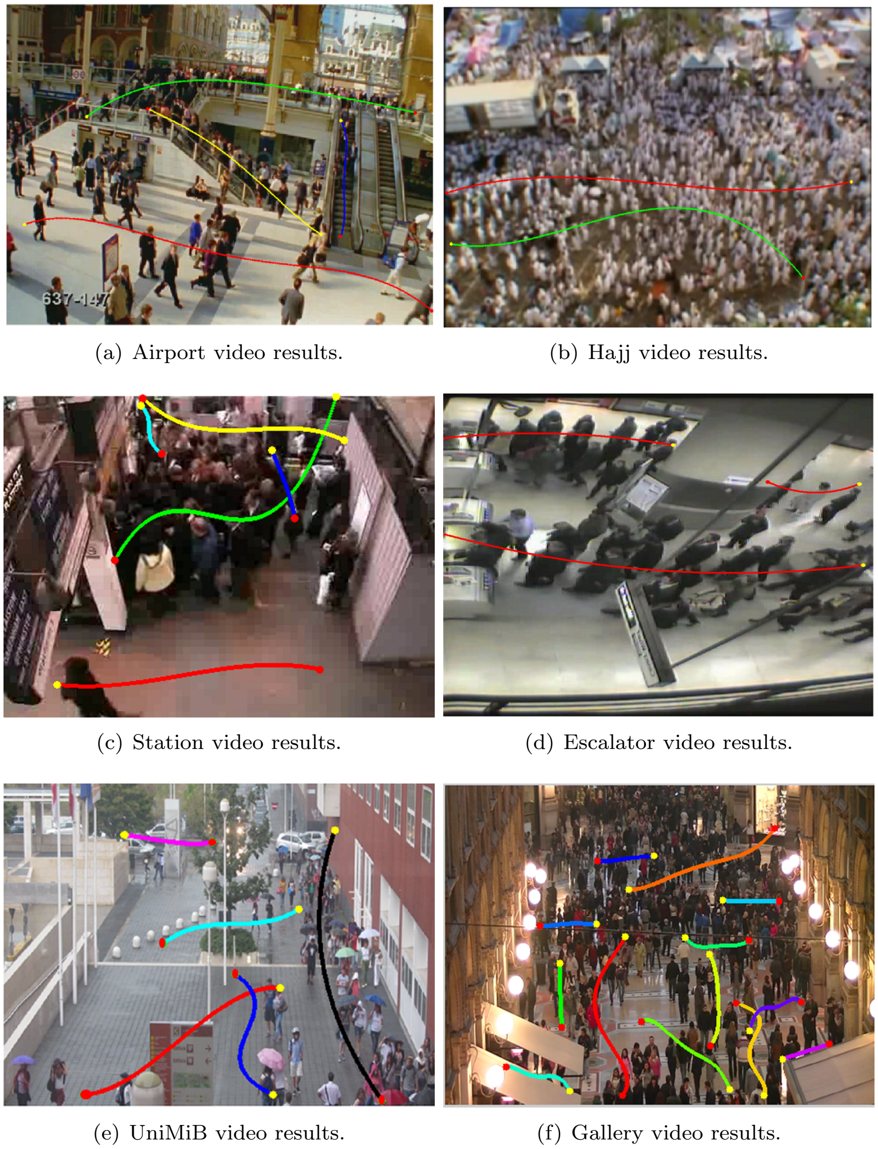
Finally, the gallery video 5.18(f) represents a rather extreme scenario that is being mostly reported to show how the proposed approach is robust to occlusions, difficult lighting conditions, high pedestrian density and lack of apparent dominant motion patterns. The scenario, in fact, should be analyzed for a longer time-frame for more interesting and substantial results, that could lead to an improved understanding of the attractiveness of shops and other potential interest points in the area.

In conclusion we can stress the fact that the proposed approach can uniformly provide very interesting results, from the perspective of characterizing dominant pedestrian flows, in all the considered crowding conditions. In the station and escalator videos the approach described in [70] accurately detects the flows but the detected tracks are not long enough to capture the whole motion information, leading to an incomplete characterization of the overall flows.

Similar considerations can be done considering the approach described in [71], in particular for the airport footage: this approach identifies small tracklets but complete information about the motion is missing while our results completely describe flows with their respective sources and sinks. In the Hajj video, moreover, the approach introduced in [71] detects redundant flows while our method correctly summarizes horizontal flow, although both approaches miss the vertical flow that the human annotator detected, as shown in Figure 5.6(b).

5.3.3 Parameters of Algorithms

Here we discuss parameters setting for proposed algorithms 1 and 2. Since both algorithms use different parameters therefore we describe parameter setting for each algorithm separately. Parameters setting for algorithm 1 is described in table 5.3.3. *step* is the first parameter, though not the actual parameter of algorithm 1, specifies the resolution of grid of particles to be overlaid on the scene. We fix the value of step to 10 for all the analyzed videos. The resolution of grid of particles for a given image of size 300x400 with step 10 is 30x40. It implies that we drop the particle at every 10^{th} pixel location while scanning from left to right (row wise) or top to bottom (column wise) of an image. We can also increase the resolution of the grid by lowering the value of *step*, but this will make the algorithm computational expensive ending up with the



(a) Airport video results.

(b) Hajj video results.

(c) Station video results.

(d) Escalator video results.

(e) UniMiB video results.

(f) Gallery video results.

Figure 5.18: Crowd flow characterization results of the proposed approach in the different scenarios.

similar results. ϵ controls the euclidean distance between the sink point of query tracklet and source point of neighbor tracklet. We set this value to 7.5 for all analyzed videos. m_t controls the matching ratio. We fix this value to 0.4 for all the analyzed videos, that implies that our algorithm accepts candidate tracklet which is at least 40% similar to that of query tracklet.

The description of parameter settings for algorithm 2 is shown in Table 5.3. Ω controls the spatial matching of any two input trajectories. Tuning of Ω is required in order to obtain semantically useful results for a given sequence, since it depends on the video frame resolution, crowd density, crowd type, i.e structured or unstructured. We set a low threshold value for gallery sequence, since it involve complex movement of people. We use higher value of Ω for structured crowds. This parameter is determined experimentally. Before running the algorithm 2 on a long video sequence, an analyst can tune this parameter to an appropriate value by observing the video for a short time. Parameter φ is the same as m_t and we set it to 0.5 for all video sequences.

Table 5.2: Parameter Setting for Algorithm 1

Variable	Description	Value
step	Control the resolution of grid	10
ϵ	allowed distance between tracklets in Algorithm 2	$step(\frac{3}{4})$
m_t	Matching ratio	0.4

Table 5.3: Parameter Setting for Algorithm 2

	Hajj	Station	Unimib	Airport	Escalator	Gallery
Resolution	384x576	360x480	360x480	360x480	480x480	360x480
Ω	120	90	90	90	120	70
φ	0.5	0.5	0.5	0.5	0.5	0.5

Some considerations must be done, finally, on the fact that the passage between local results, related to a single video segment, and final overall global motion flow description does not employ temporal information associated to the local flows (which would probably be necessary for a proper tracking algorithm). The example shown in Figure 5.17 shows that the clustering technique devised and adopted for this final passage actually allows considering all flows that, even just temporarily (i.e. not in all segments), represented a relevant and noticeable flow of pedestrians. Moreover, the proposed approach actually exploits the fact that pedestrians tend to follow similar paths in the environment, sometimes imitating the movement of other pedestrians: the trajectory completion function described in Algorithm 2, in fact, supports the detection of an overall pedestrian flow even in a single video segment, even though a single pedestrian would not be able to cover it, as long as other pedestrians are moving along a similar path.

Chapter 6

Detection of Social Groups in Pedestrian Crowds

6.1 Introduction

Crowded scenes are composed of large number of people exhibiting different behaviors in a constrained environment. The analysis of the behavior of pedestrians and crowds in video surveillance systems is a topic of growing interest supporting an improved understanding of human behavior and decision making activities through several functions like activity recognition [72], automated analysis of the flow of large crowds, for example through crowd flow segmentation and crowd counting [73], the discovery of frequent pathways [74], the identification of crowd behaviors [67] and abnormal event detection [75, 76]. All these studies either focus on individuals or on the overall crowd, considered as large set of pedestrians, not considering the importance of some social interaction among pedestrians: most pedestrians do not really walk alone [77], and researchers observed in most situations pedestrians actually walk in groups. Some interesting forms of social interaction and adaptive behaviors can be observed at the group level and they are growingly investigated in the area of pedestrian and crowd modeling and simulation [78, 77]. On the other hand, detecting and analyzing social groups of people is still a less studied topic.

A few recent works [79, 80] are aimed at the detection of groups without using future information about the dynamics of the scene. [79] employed Decentralized Particle Filtering (DPF) for group detection while [80] employed unsupervised group detection method based on Dirichlet Process Mixture Model (DPMM) which exploits proxemics to determine group formation. Other approaches like [81, 82, 83] use social forces to analyze motion patterns and recognize groups. These social forces based methods are based on pairwise similarity between trajectories of pedestrians followed by a clustering phase. An approach described in [84] extracts trajectory information from the whole video, then trajectories are temporally analyzed in order to determine the affiliation of each

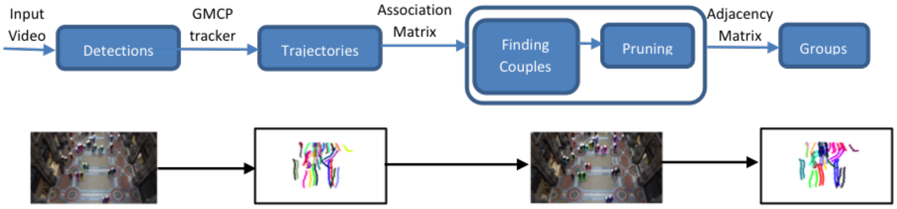


Figure 6.1: Proposed Methodology for Group Detection.

pedestrian to a particular group. Pedestrians are grouped in a bottom-up fashion by employing hierarchical clustering using pairwise proximity and velocity. In [85], both spatial locations and velocities are used within a modified Hausdorff distance to compute trajectory similarities. In [86], Euclidean distance metric is used to cluster vehicle trajectories. [74] measures trajectory similarities using Longest Common Sub-Sequence. [87, 88] use Hausdroff and Dynamic Time Warping metric to measure trajectory similarities. The problems with employing all above pairwise similarity measures are that they are computational expensive and lack probabilistic explanation. On the other hand, instead, recent works are focusing on modeling the distribution of trajectories locations and velocity observations [89, 90].

The approach presented in this paper starts by extracting trajectory information from the whole video and building an *Association Matrix* that captures the joint probability distribution of start and stop locations of all pedestrians to all other pedestrians in the scene and it adopts a bottom-up hierarchical clustering approach similar to the one adopted in [84] to discover social groups. The main contributions of the work are: (i) instead of considering whole trajectories, we consider only two points (start and stop) making the overall group detection process computationally less expensive and more suitable for real-time operation; (ii) our approach does not require training; (iii) the usage of *Association Matrix* for discovering couples and *Adjacency Matrix* for discovering groups; (iv) Our approach requires only one parameter setting.

The paper is organized as follows: in the following we present the overall proposed approach, while Sect. 6.3 describes the clustering algorithm. Section 6.4 describes the achieved experimental results, also by comparing the proposed approach with the most relevant existing alternatives. Conclusions and future developments end the paper.

6.2 Proposed Methodology

The overall framework for automatic detection of pedestrian social groups in crowds is described in Fig. 6.1; the input is a video sequence in which individual pedestrians are detected: we adopted a semi-automated approach for detecting pedestrians, however, any detector could be used. The second phase is associated to the tracking of the detected pedestrians: once again, we adopted a specific approach, but in principle any tracker could be used. Pedestrians detected in first frame are tracked through multiple frames using the Generalized Minimum Clique Graphs (GMCP) [51] method, which is aimed to solve the data association problem by exploiting both motion and appearance in a global manner. The input to GMCP is a graph, in which all the detections in each frame are connected to all other detections in all other frames. The output is the set of several subgraphs, in which the detections belonging to common entities are connected. The trajectory of pedestrian in the scene is a set of tuples (x, y, t), where x and y are the horizontal and vertical coordinates of the location at time t. Therefore, the trajectory of pedestrian is represented by $\{(x_1, y_1, t_1), \dots (x_n, y_n, t_n)\}$, where n is total length of the trajectory of a pedestrian over a time window T. Once the trajectories are extracted, the next step is to construct an *Association Matrix*, that captures the joint distribution of source and sink locations of all pedestrians to all other pedestrians in the scene.

The first intuition behind this approach is that pedestrians appear and disappear at relatively precise and recurring locations, such as doors, gateways or particular portions of the edges of the scene in videos taken from the fixed camera. We refer to locations where pedestrian appear as *sources* (potential origin of a trajectory) and locations where they disappear as *sinks* (potential destinations of a trajectory). The second consideration is that pedestrian crowd motion is driven by adaptive processes based on local interactions among pedestrians; the latter are more stable and stronger when they move in a group, such as friends or family members, since these individuals exhibit more coherent movements. There are two key characteristics of the members of group: (i) the spatio-temporal relationships of members of group tend to remain stable over time, with members preserving a small distance from one another and avoiding separation unless an obstacle comes in a way; (ii) the velocities of group members are also correlated. We capture the above two characteristics by building an *Association Matrix*. The key notion of our approach is to cluster pedestrians having similar distributions across the source and sink locations.

In order to build an *Association Matrix*, we assume two discrete jointly distributed random variables $\mathbf{X}$, representing "source" locations of the trajectories and $\mathbf{Y}$ representing "sink" locations, where $\mathbf{X} \in \mathbf{\Omega}$ and $\mathbf{Y} \in \mathbf{\Omega}$ and set $\mathbf{\Omega}$ represents all the possible scene locations. Let $\mathbf{X}$ and $\mathbf{Y}$ take values in the sets $\{x_1, x_2, \dots, x_n\}$ and $\{y_1, y_2, \dots, y_n\}$ respectively, where x_k and y_k are the start and stop locations of pedestrian's trajectory k. An *Association Matrix* for n trajectories is shown below.

$$P(X, Y) = \begin{Bmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1n} \\ p_{21} & p_{22} & p_{23} & \cdots & p_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & p_{n3} & \cdots & p_{nn} \end{Bmatrix}$$

Each row/column of an *Association Matrix* shows the probability distribution of the source and sink points of single pedestrian P_k over all other n pedestrians in the scene. Let P_k is the distribution of sources and sinks of pedestrian trajectory k with all other n pedestrian trajectories and represented as $\{p_{k1}, p_{k2}, p_{k3}, \ldots, p_{kn}\}$, where $p(k, j)$, is the joint probability of start and stop locations for pedestrian trajectories k and any pedestrian trajectory j. For pedestrian trajectory k, we use a Gaussian likelihood model [112] to compute its probability of its starting from start location of other trajectory j in the scene as equation 6.1

$$p_x(k, j) = \mathrm{e}^{-\| \frac{x_k - x_j}{\sigma} \|} \tag{6.1}$$

Where x_k is the source location of trajectory k and x_j is the source location of trajectory j. Similarly, probability of stopping for trajectory k from stop location of trajectory j in the scene as equation 6.2

$$p_y(k, j) = \mathrm{e}^{-\| \frac{y_k - y_j}{\sigma} \|} \tag{6.2}$$

Where y_k is the sink location of trajectory k and y_j is the sink location of trajectory j. Assuming independence among the trajectories, we multiply $p_x(k, j)$ and $p_x(k, j)$ to calculate joint probability $p(k, j)$ for pedestrian trajectories k and j. In the same way, we compute joint probabilities of all other trajectories and after normalization, we obtain an *Association Matrix.* With the detection of new pedestrians, new trajectories are extracted and matrix is updated in the same way.

Association matrix help us in capturing the walking behavior of pedestrians. A single pedestrian (not member of a group) tends to move or stop freely in the environment, changing his/her speed and keep a distance from other pedestrians or obstacles, pursuing is/her own goals. This behavior uniquely identifies his/her source and sink locations. Member of a group generally move and stop together following the notion of group *entitativity* [113], which defines Gestalt psychology of common fate, similarity in appearance and behavior, proximity, and pregnance (patterning). In other words, to a certain extent, a group can be considered as a single entity, as a whole in the environment like, other single pedestrians. Therefore, members of group produce similar distributions and this could be easily detected by looking at the above defined *Association Matrix.* In the next step, we illustrate clustering algorithm that take *Association Matrix* as input.

6.3 Bottom up Hierarchical Clustering

We adopt a bottom-up hierarchical clustering approach which is a three step process. In the first step, we assign distinct cluster identifiers by treating each pedestrian as a separate cluster. In the second step, our clustering algorithm discovers couples by measuring the difference between distribution of each pedestrian with the distribution of all other pedestrians in the scene by using *Kullback-Leibler (KL) divergence*, also known as relative entropy, denoted by $D_{KL}(P_r||P_k)$, computed by using equation 6.4 and selects the one that minimizes equation 6.3. For example, to find a group partner for a pedestrian with distribution P_r, we select a pedestrian with distribution P_k that minimizes the equation 6.3.

$$\operatorname*{argmin}_{k=1}^{n}(D_{KL}(P_r||P_k)) \tag{6.3}$$

$$D_{KL}(P_r||P_k) = \sum_i P_r(i) \ln \frac{P_r(i)}{P_k(i)} \tag{6.4}$$

This process always proposes for each pedestrian the best possible partner to form a couple, although this candidate partner might even be a bad partner, since the pedestrians do actually not follow similar paths in terms of source and sink in their trajectories. The next step is thus to prune these bad couples. Couples are labeled as bad if the joint probability $p(r,k)$ (computed in section 6.2), is greater than a specified threshold τ_s.

After pruning, an adjacency matrix is generated which captures the connectivity information among all pedestrians. In order to illustrate the situation with an example, consider the 6 x 6 matrix below that captures the connectivity information among six pedestrians. In matrix A '1' represents an edge between two pedestrians while '0' shows that there is no edge exits between them. As shown in matrix, pedestrian p_1 is adjacent to p_2, p_2 is adjacent to p_4 and p_1, p_3 to p_5, p_4 to p_2 while p_6 is not connected with any other pedestrian.

$$A = \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \end{array} \begin{array}{c} \begin{array}{cccccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

In the third step of the algorithm, group of couples, those having strong intergroup closeness are merged into a larger group e.g, $G(p_1,p_2)$ have a strong intergroup closeness with $G(p_2,p_4)$ because these two group of couples have one member in common i.e, p_2. Pseudocode of the third step (algorithm 3) automatically discovers groups of pedestrian by taking adjacency matrix as input.

One could take a top-down approach by considering the entire crowd as one group and iteratively splitting into subgroups. We choose the bottom-up approach because it is more efficient in the situations where the crowd is composed of small groups (and this represents the most frequent situation, according to empirical observations [77]). Our clustering algorithm does not require a predefined number of clusters as compared to other traditional clustering algorithms e.g, K-means or spectral clustering. Our algorithm automatically discovers the number of groups by constructing a connectivity graph among pedestrians having similar distributions.

Algorithm 3 Discovering intergroup closeness and agglomerating couples from adjacency matrix

```
Input: Adjacency Matrix $A$
Output: Groups G
initialize discovery vector $D$ equal to number of pedestrian to zeros
initialize group $G$ cluster.
idx = 1                                          ▷ ID of the group(cluster)
for all pedestrians $n$ do
    if $n$ is not discovered then
        D(n) = 1
        insert $n$ in $G[idx]$
        $ptr = 1$
        while $ptr \leq length(G[idx])$ do
            find neighbor $\hat{n}$ of $n$ in $A$
            if D($\hat{n}$)= 0 then
                D($\hat{n}$)= 1
                insert $\hat{n}$ in $G[idx]$      ▷ insert $\hat{n}$ in group(cluster)ID $idx$
                $n = \hat{n}$                     ▷ update $n$
            end if
            increment $ptr$ by 1
        end while
    end if
    increment $idx$ by 1
end for
```

Table 6.1: Details of datasets: Key-ppf: people per frame

	ETH	HOTEL	GALLERY	SU2-L	SU2-H
Total number of people	360	390	685	639	2678
Number of groups	74	59	85	127	410
Average number of ppf	6	8	12	17	50
Number of Frames	1448	1168	1002	600	600

6.4 Experimental Results

This section presents both quantitative and qualitative analyses of the results obtained from experiments. We carried out our experiments on a PC of 2.6 GHz (Core i5) with 4.0 GB memory, running a Mathlab implementation of the presented algorithm. We validate our proposed group detection approach on video sequences made available from other research groups and acquired through field observations. The overall set of video includes situations including both the so called structured and unstructured crowds with different density conditions. The videos named as *eth* and *hotel* from [114] are recorded in low density situations, *su2* from [84] consists of two 15 minutes sequences, *su2l* and *su2h*. The first sequence, *su2l* has 10-20 pedestrians per frame and covers low density situations, while the second sequence, *su2h*, has more than 50 pedestrians per frame and covers high density situations. The dataset, *gallery* from [108] is recorded in a relatively high density situations. Table 6.1 shows the summary of each analyzed video.

In order to evaluate our proposed approach, we obtain ground truth of all video sequences: a human observer watched a version of video with IDs overlaid on individuals in the scene. For quantitative evaluation of our proposed grouping method, we first compare ground truth and auto-estimated group size for each pedestrian.

Some considerations must be done about the application of the presented group detection approach before discussing results. The overall workflow described in Fig. 6.1 intrinsically implies a *time window* in which the pedestrians are identified and tracked, in which their sources and sinks are identified and in which all the group detection mechanisms are applied. Therefore, the presented approach produces results that are to be considered valid *within* a time window. The length of the time window, with respect to both general pedestrian dynamics and the overall travel time within a given scene, significantly influences the performance of the group detection: in particular, too short time windows make it difficult to actually perceive differences between genuine groups and simple pedestrians that, for a short time, move in a very similar way, that would represent false positives. A larger time window, close to the length of the average travel time of pedestrians in the scene, would intuitively improve group detection, but it would also reduce the *frequency* of the detection of groups in the scene. While facing this problem is object of current and future works, we will discuss the effect of the choice of a time window size on the precision of group

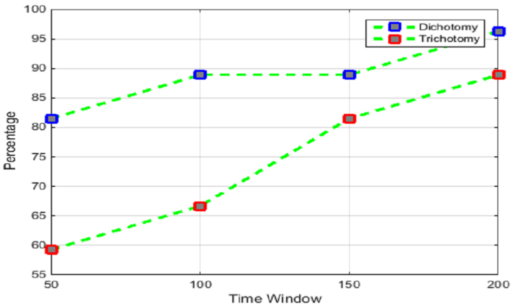


Figure 6.2: Effect of Time window on accuracy

detection.

As previously mentioned, the trajectories were extracted by means of a semi automatic pedestrian detection mechanism; this fact should not hinder or favor the proposed approach compared to existing ones, but this is subject to further analyses. To better discuss the accuracy of the group detection we categorize a member of a group under two coding schemes: *Dichotomous* coding scheme and *Trichotomous* coding scheme. In the former, we checked whether a pedestrian is alone or in group while in the trichotomous coding scheme we determine the size of the group. We compare these coding schemes with the ground truth and from the experiments, we observe that our proposed grouping algorithm achieves 93.6% accuracy in dichotomous while 88.2% accuracy in trichotomous coding scheme on average for all the analyzed videos. As shown in Table 6.2, the performance of our proposed approach for the first two videos, i.e, *eth* and *hotel*, under dichotomous and trichotomous coding scheme is very high, since these videos covers low density situations, where pedestrians are relatively distant one from another, while in the other videos, where the density is relatively high and pedestrians are moving at a short distance, the performance of our algorithm decreases. Further investigations on video sequences reveals that the performance of our proposed approach gradually decreases with the increase in crowd density in general, irrespectively of the adopted video: in high density situations, the available space around the group is reduced and this forces the configuration of the group to change to adapt to the contextual situation, voiding the assumptions behind our approach[1].

From the experiments, we also analyzed how the accuracy of the proposed approach is influenced by the size of the time window. If the time window is shorter than maximum travel time of pedestrian in the scene, this implies that pedestrian trajectories are analyzed for a short duration and the algorithm may lead to false positives. In such situations, two pedestrians walking close to each other for a short period will be detected as couple, although they cease to move

[1] It must be noted that, however, empirical data about the proxemic behavior of groups in relatively high density situation is still lacking, therefore we do not have clear idea of how groups behave in this kind of situation.

Table 6.2: Dichotomy and Trichotomy on different video sequences

Videos	τ_s	**Dichotomoy**	**Trichotomy**
ETH	10^{-10}	100.0%	95.0%
HOTEL	10^{-15}	100.0%	92.0%
GALLERY	10^{-19}	96.3%	89.9%
SU2-L	10^{-12}	90.70%	86.70%
SU2-H	10^{-12}	81.06%	77.24%

together if analyzed for a longer duration. Figure 6.2, shows the accuracy of dichotomous and trichotomous coding scheme with varying time window for the *gallery* video sequence. In this video sequence, maximum travel time of pedestrian is 200 frames. As it is obvious from Figure that accuracy of our grouping algorithm increases with the increase in duration of time window. Since in general this value is unknown, we can consider a reasonable initialization value to be set according to known data such as the average pedestrian walking speed and the dimension of the observed area

Comparison with state-of-the-art

We compare our proposed grouping algorithm with the ones that are closest to the present approach, respectively described in [84] and [115]. In [84], the researchers identify small group of pedestrians by combining spatial proximity and velocity cues into a pairwise distance computed for the whole trajectory. Intergroup closeness between two groups of pedestrian is measured by symmetric Hausdorff distance. They construct a connectivity graph and adopted bottom-up hierarchical clustering approach that start by treating each individual as separate cluster and gradually discovers large groups by merging two clusters that satisfy intergroup closeness. In contrast, our proposed algorithm does not compute pairwise distance measure for the whole trajectory, instead, we consider only the source (start point of trajectory) and sinks (stop point of trajectory) points of any two trajectories. In order to show the effectiveness of our proposed approach under dichotomous and trichotomous, we compare our approach with [84] using *su2* video sequence as shown in Table 6.3.

In order to further discuss the quality of the achieved results, we quantitatively compare our approach with [84] in terms of time complexity and shown in Figure 6.3. The horizontal axis of the Figure shows the increasing mean length of two trajectories belonging to pedestrian couples. Pedestrian couple with

Table 6.3: Comparison with state-of-the-art method

Data set	Proposed		[84]	
	Dichotomy	Trichotomy	Dichotomy	Trichotomy
SU2-L	**90.70%**	**86.70%**	84.00%	75.00%
SU2-H	**81.06%**	**77.24%**	75.00%	72.00%

mean length of 200 have longer trajectories than pedestrian couple with mean length of 50. The vertical axis of the Figure shows total computational time. The overall computational time is significantly less than [84], for which the cost increases linearly with the increase in length of trajectories, since method [84] computes pairwise distance between all points of the trajectories while in our case the computational time is constant, since we consider only the source and sink point of the trajectories. Our approach is also based on bottom-up hierarchical clustering that also starts by treating each pedestrian as a separate cluster. In the second step, our algorithm tries to find couples and construct a connectivity graph. In the third step, instead of measuring symmetric Hausdorff distance as in [84] (which is widely used for shape matching and trajectory analysis) for merging two groups, we merged two groups of couples into larger group by adopting Algorithm 3.

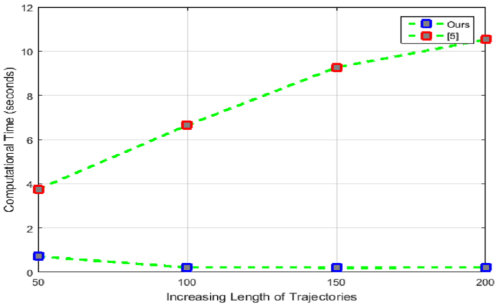
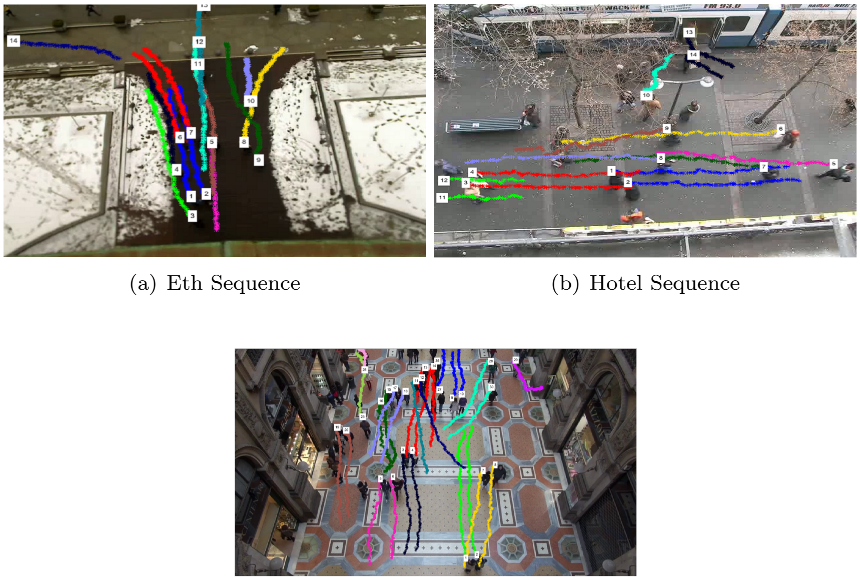


Figure 6.3: Time Complexity

In [115], instead, the authors proposed a structural SVM-based learning framework that extract Hall's proxemics and Granger's causality as main features from trajectories of pedestrians, then a supervised bottom-up hierarchical clustering approach to discover groups of pedestrians was adopted. This method is very effective and it implied the adoption of an original machine learning method characterized by a plausible model of pedestrian behavior, but it is also computationally expensive, requiring off-line data for learning and training. In contrast, our approach is a relatively simple three step process, it does not require off-line training and learning which makes it suitable for real-time applications, granted that proper pedestrian detection and tracking algorithms are adopted. Figure 6.4 reports visual examples of our proposed algorithm.



(a) Eth Sequence

(b) Hotel Sequence

(c) Gallery Sequence

Figure 6.4: Qualitative results of different video sequences

Chapter 7

Summary and Future Work

The main theme of this thesis is two fold,i.e, understanding crowd dynamics in videos of (i), high density crowds and (ii) low density crowds. Typical examples of high density crowds include marathons, religious festivals while malls, airports, subways etc covers low dense situations. In this thesis, I adopt different approaches in order to deal with different kinds of problems coming from these two categories of crowd. This thesis highlights the problems of crowd flow segmentation, counting and crowd behaviour understanding problems originating from the first category and social group detection from the other category.

In particular, first part of the thesis, I adopt holistic approach to generate a global representation of the scene that captures both dynamics of the crowd and structure of the scene. This was achieved by extracting global features, i.e optical flow from the scene. For the crowd flow segmentation problem, the optical flows vectors are clustered by using K-means clustering followed by the blob absorption approach. Using the segmentation information, we continue to estimate the number of people in each segment by carrying out the blob analysis and blob size optimization approach.

In the second part of the thesis, I use trajectory information by building an association matrix that captures the joint probability distribution of the source and sink of one trajectory with all other trajectories of pedestrians in the scene. In particular category, we extract the pedestrian trajectory by using detection and tracking approach.

I summarize the main contribution of this thesis in the following section

7.0.1 Summary of Contributions

Crowd Flow Segmentation and Crowd Counting

In this framework, we have considered both high and low density crowds and proposed a framework that automatically detects dominant motion flows and counts the number of people in each flow. Such kind of analysis provides a useful input to pedestrian simulation models. A first employment of the our analysis is related to the actual initial configuration of the simulation scenario. Second way to exploit data resulting from automated video analysis is represented by pedestrian counting and density estimation: the indication of the average number of pedestrians present in the simulated portion of the environment is important in configuring the start areas. Finally, we can use the above analysis in the validation of simulation results. Our approach is applicable in many different situations and it is independent of local conditions and camera viewpoints.

1. Combining crowd flow segmentation and crowd counting in one framework,
2. Adoption of global features, optical flow, to segment the crowd,
3. Blob absorption approach that improve the state-of-the-art results
4. Estimating number of people in each segment
5. Proposed framework does not require local features like detection and tracking
6. No training is required

Crowd behaviour understanding

The framework presents automated analysis of videos in naturalistic conditions and the identification of points of entrance (sources) and exit (sinks) of the most significant pedestrian flows. The approach adopts optical flow for the identification of pedestrian movements, and it considers the analyzed video as a set of sequences. The latter are analyzed separately, producing tracklets that are then clustered to produce global trajectories, defining both sources and sinks, but also characterizing the movement of pedestrians in the scene. The algorithms work according to geometric considerations essentially considering the plausibility of extending tracklets associated to optical flow by connecting them when they represent a smooth continuation one of another, and then clustering those sharing a significant subsequence.

The approach has been presented in details, also setting it in the current state of the art. Results of its application to the analysis of videos made available by other researchers and by our research group have been discussed mainly with reference to two aspects: (i) the capability of producing long and dense tracks associated to pedestrian movements, also with reference to the most relevant approaches present in the computer vision literature, and (ii) the capability of summarizing pedestrians' movements, identifying at the same time sources and sinks. For both aspects, the considered videos cover a wide range of crowding situations, from medium-low to relatively high crowding conditions, in cases of structured and unstructured crowds.

1. generating dense and long trajectories,
2. identifying sources and sinks,

3. understanding behaviour of the crowd in the scene by considering full length video,

4. achieve the above results without requiring object detection, tracking, nor training, targeting employment in naturalistic conditions.

5. introduction of new metric systems for evaluating the proposed and other like frameworks

Social group detection

In this framework, We propose a novel approach for automatic detection of social groups of pedestrians in crowds by considering only start (source) and stop (sink) locations of pedestrian trajectories. We build an *Association Matrix* that captures the joint probability distribution of starts and stops locations of all pedestrian trajectories to all other pedestrian trajectories in the scene. Pedestrians exhibiting similar distribution are combine in a group, where as similarity among the distributions is measuread by $Kullback - Leibler(KL) divergence$. We adopt bottom-up hierarchical clustering approach, which is three step process. In first step, we treat all the individuals as independent clusters, In the second step, couples are detected and after pruning of bad couples, *AdjacencyMatrix* is generated. Later on, in step three, using the *AdjacencyMatrix*, groups of couples, those have strong intergroup closeness (similarity) are merged into a larger group.

1. instead of considering whole trajectories, we consider only two points (start and stop) making the overall group detection process computationally less expensive and more suitable for real-time operation,

2. our approach does not require training,

3. the usage of *Association Matrix* for discovering couples and *Adjacency Matrix* for discovering groups,

4. our approach requires only one parameter setting..

7.0.2 Future Work

The methods developed in this thesis can be improved and extended in the following ways

Crowd flow segmentation and crowd counting

For detail understanding of crowd behaviour, the method proposed in this thesis can be extended by labelling different segments with one of commonly observed behaviours. The common type of behaviours observed in these situations are: lane formation, bottlenecks, fountain (people spreading in different direction from one point), merging and splitting etc. Once the labels are assigned to different segments will generate a much higher representation of the scene that will be much easier for the human operator to understand and react to the situation.

Another extension to current the crowd counting framework is to rely less on motion information and rely more on image content. Because motion information is extracted by computing optical flow which will affect the accuracy of crowd counting framework in case of severe changes in lightening conditions. This could be performed by segmenting the image content into crowd and non-crowd regions. This can be achieved by extracting appearance base features like SIFT and utilization of SIFT words or by texture analysis by using local binary pattern, GLCM, or Fourier spectrum analysis of the image.

Crowd behaviour understanding

The present framework can be improved and extended in four directions:

1. extensions of the approach to produce information that can be more directly used by modelers for the configuration of simulation scenario, that is, origin destination matrices and traffic assignments: this point will require a quantitative characterization of the sources, sinks and main flows, and it will also imply a different form of experimentation analyzing longer videos, but the quantitative evaluation of this development is rather straightforward;

2. extension of the approach to consider multi-camera scenarios: the present approach is very promising but so far we did not analyze large scenarios in which the analyzed area can only be covered by more cameras and scenes; this will require a higher level of correlation among results of the application of the same approach to different videos, a higher level that is also related to the next point;

3. extension of the approach to perform a semantic analysis of the results, for the identification and treatment of situations essentially characterizable as (i) confluence of different flows in a single one, (ii) separation of initially joint flows; these situations, as well as the previous one, will likely require the adoption of some form of knowledge representation and reasoning on graph-like structures associated either to static spatial representations or to the results of the application of the proposed approach.

4. extension of the approach to multi-target object tracking as in [116] but the target data set used comprise low density scenarios. Our target is to extend the approach to more dense and complex videos.

Social group detection

The present framework for the group detection on one side, can be extended by improving the accuracy of the group detection and, on the other, at clarifying the overall possibility to apply this method (i) respecting real-time constrains or simply (ii) for providing significant data supporting pedestrian modelling and simulation within an integrated approach. Th accuracy of the present framework is coupled with performance of pedestrian detection and tracking, which in many real time cases, difficult due to severe occlusions and clutter. The present framework can be extended by extracting features trajectories instead of detecting and tracking individuals, i.e , KLT from the scene. The KLT trajectories related to a group can be clustered by adopting source and sink co-clustering algorithm. The co-clustering algorithm will cluster trajectories which follows the similar distribution across the source and sink locations.

Bibliography

[1] J. C. S. J. Junior, S. R. Musse, C. R. Jung, Crowd analysis using computer vision techniques, IEEE Signal Processing Magazine 27 (5) (2010) 66–77.

[2] M. Rodriguez, S. Ali, T. Kanade, Tracking in unstructured crowded scenes, in: Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 1389–1396.

[3] B. Zhan, D. N. Monekosso, P. Remagnino, S. A. Velastin, L.-Q. Xu, Crowd analysis: a survey, Machine Vision and Applications 19 (5-6) (2008) 345–357.

[4] M. Butenuth, F. Burkert, F. Schmidt, S. Hinz, D. Hartmann, A. Kneidl, A. Borrmann, B. Sirmacek, Integrating pedestrian simulation, tracking and event detection for crowd analysis, in: Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on, IEEE, 2011, pp. 150–157.

[5] A. Marana, S. Velastin, L. Costa, R. Lotufo, Estimation of crowd density using image processing, in: Image Processing for Security Applications (Digest No.: 1997/074), IEE Colloquium on, IET, 1997, pp. 11–1.

[6] A. C. Davies, J. H. Yin, S. Velastin, et al., Crowd monitoring using image processing, Electronics & Communication Engineering Journal 7 (1) (1995) 37–47.

[7] D. Kong, D. Gray, H. Tao, A viewpoint invariant approach for crowd counting, in: Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, Vol. 3, IEEE, 2006, pp. 1187–1190.

[8] A. N. Marana, M. A. Cavenaghi, R. S. Ulson, F. Drumond, Real-time crowd density estimation using images, in: Advances in visual computing, Springer, 2005, pp. 355–362.

[9] A. N. Marana, L. da Fontoura Costa, R. Lotufo, S. A. Velastin, Estimating crowd density with minkowski fractal dimension, in: Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on, Vol. 6, IEEE, 1999, pp. 3521–3524.

[10] L. Xiaohua, S. Lansun, L. Huanqin, Estimation of crowd density based on wavelet and support vector machine, Transactions of the Institute of Measurement and Control 28 (3) (2006) 299–308.

[11] H. Rahmalan, M. S. Nixon, J. N. Carter, On crowd density estimation for surveillance.

[12] C. Regazzoni, A. Tesei, V. Murino, A real-time vision system for crowding monitoring, in: Industrial Electronics, Control, and Instrumentation, 1993. Proceedings of the IECON'93., International Conference on, IEEE, 1993, pp. 1860–1864.

[13] S.-Y. Cho, T. W. Chow, C.-T. Leung, A neural-based crowd estimation by hybrid global learning algorithm, Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 29 (4) (1999) 535–541.

[14] S.-Y. Cho, T. W. Chow, A fast neural learning vision system for crowd estimation at underground stations platform, Neural processing letters 10 (2) (1999) 111–120.

[15] T. W. Chow, J.-F. Yam, S.-Y. Cho, Fast training algorithm for feedforward neural networks: application to crowd estimation at underground stations, Artificial intelligence in engineering 13 (3) (1999) 301–307.

[16] R. Ma, L. Li, W. Huang, Q. Tian, On pixel count based crowd density estimation for visual surveillance, in: Cybernetics and Intelligent Systems, 2004 IEEE Conference on, Vol. 1, IEEE, 2004, pp. 170–173.

[17] D. Roqueiro, V. A. Petrushin, Counting people using video cameras, The International Journal of Parallel, Emergent and Distributed Systems 22 (3) (2007) 193–209.

[18] A. B. Chan, N. Vasconcelos, Bayesian poisson regression for crowd counting, in: Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 545–551.

[19] A. B. Chan, M. Morrow, N. Vasconcelos, Analysis of crowded scenes using holistic properties, in: Performance Evaluation of Tracking and Surveillance workshop at CVPR, 2009, pp. 101–108.

[20] A. B. Chan, N. Vasconcelos, Counting people with low-level features and bayesian regression, Image Processing, IEEE Transactions on 21 (4) (2012) 2160–2177.

[21] A. B. Chan, Z.-S. J. Liang, N. Vasconcelos, Privacy preserving crowd monitoring: Counting people without people models or tracking, in: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, IEEE, 2008, pp. 1–7.

[22] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, Vol. 1, IEEE, 2005, pp. 886–893.

[23] P. Felzenszwalb, D. McAllester, D. Ramanan, A discriminatively trained, multiscale, deformable part model, in: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, IEEE, 2008, pp. 1–8.

[24] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, Pattern Analysis and Machine Intelligence, IEEE Transactions on 32 (9) (2010) 1627–1645.

[25] D. Conte, P. Foggia, G. Percannella, F. Tufano, M. Vento, Counting moving people in videos by salient points detection, in: Pattern Recognition (ICPR), 2010 20th International Conference on, IEEE, 2010, pp. 1743–1746.

[26] D. Conte, P. Foggia, G. Percannella, F. Tufano, M. Vento, A method for counting people in crowded scenes, in: Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on, IEEE, 2010, pp. 225–232.

[27] N. Dong, F. Liu, Z. Li, Crowd density estimation using sparse texture features, Journal of Convergence Information Technology 5 (6) (2010) 125–137.

[28] W. Ma, L. Huang, C. Liu, Advanced local binary pattern descriptors for crowd estimation, in: Computational Intelligence and Industrial Application, 2008. PACIIA'08. Pacific-Asia Workshop on, Vol. 2, IEEE, 2008, pp. 958–962.

[29] W. Ma, L. Huang, C. Liu, Crowd estimation using multi-scale local texture analysis and confidence-based soft classification, in: Intelligent Information Technology Application, 2008. IITA'08. Second International Symposium on, Vol. 1, IEEE, 2008, pp. 142–146.

[30] W. Ma, L. Huang, C. Liu, Crowd density analysis using co-occurrence texture features, in: Computer Sciences and Convergence Information Technology (ICCIT), 2010 5th International Conference on, IEEE, 2010, pp. 170–175.

[31] T. Zhao, R. Nevatia, Bayesian human segmentation in crowded situations, in: Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on, Vol. 2, IEEE, 2003, pp. II–459.

[32] B. Wu, R. Nevatia, Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors, in: Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on, Vol. 1, IEEE, 2005, pp. 90–97.

[33] P. H. Tu, J. Rittscher, Crowd segmentation through emergent labeling, in: Statistical Methods in Video Processing, Springer, 2004, pp. 187–198.

[34] G. J. Brostow, R. Cipolla, Unsupervised bayesian detection of independent motion in crowds, in: Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, Vol. 1, IEEE, 2006, pp. 594–601.

[35] X. Huang, L. Li, T. Sim, Stereo-based human head detection from crowd scenes, in: Image Processing, 2004. ICIP'04. 2004 International Conference on, Vol. 2, IEEE, 2004, pp. 1353–1356.

[36] D. Faulhaber, H. Niemann, P. Weierich, Detection of crowds of people by use of wavelet features and parameter free statistical models., in: MVA, 1998, pp. 286–289.

[37] L. Dong, V. Parameswaran, V. Ramesh, I. Zoghlami, Fast crowd segmentation using shape indexing, in: Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on, IEEE, 2007, pp. 1–8.

[38] A. Fod, A. Howard, M. J. Mataric, A laser-based people tracker, in: Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on, Vol. 3, IEEE, 2002, pp. 3024–3029.

[39] H. Zhao, R. Shibasaki, A novel system for tracking pedestrians using multiple single-row laser-range scanners, Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on 35 (2) (2005) 283–291.

[40] J. Cui, H. Zha, H. Zhao, R. Shibasaki, Laser-based detection and tracking of multiple people in crowds, Computer Vision and Image Understanding 106 (2) (2007) 300–312.

[41] S. Ali, M. Shah, A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis, in: Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, IEEE, 2007, pp. 1–6.

[42] O. Ozturk, T. Yamasaki, K. Aizawa, Detecting dominant motion flows in unstructured/structured crowd scenes, in: Pattern Recognition (ICPR), 2010 20th International Conference on, IEEE, 2010, pp. 3533–3536.

[43] G. Eibl, N. Brändle, Evaluation of clustering methods for finding dominant optical flow fields in crowded scenes, in: Pattern Recognition, 2008. ICPR 2008. 19th International Conference on, IEEE, 2008, pp. 1–4.

[44] S. Srivastava, K. K. Ng, E. J. Delp, Crowd flow estimation using multiple visual features for scenes with changing crowd densities, in: Advanced Video and Signal-Based Surveillance (AVSS), 2011 8th IEEE International Conference on, IEEE, 2011, pp. 60–65.

[45] H. Ullah, N. Conci, Crowd motion segmentation and anomaly detection via multi-label optimization, in: ICPR workshop on Pattern Recognition and Crowd Analysis, 2012.

[46] W. Li, J.-H. Ruan, H.-A. Zhao, Crowd movement segmentation using velocity field histogram curve, in: Wavelet Analysis and Pattern Recognition (ICWAPR), 2012 International Conference on, IEEE, 2012, pp. 191–195.

[47] T. Zhao, R. Nevatia, Tracking multiple humans in crowded environment, in: Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, Vol. 2, IEEE, 2004, pp. II–406.

[48] Z. Khan, T. Balch, F. Dellaert, An mcmc-based particle filter for tracking multiple interacting targets, in: Computer Vision-ECCV 2004, Springer, 2004, pp. 279–290.

[49] C. Hue, J.-P. Le Cadre, P. Perez, Posterior cramer-rao bounds for multi-target tracking, Aerospace and Electronic Systems, IEEE Transactions on 42 (1) (2006) 37–49.

[50] D. Sugimura, K. M. Kitani, T. Okabe, Y. Sato, A. Sugimoto, Using individuality to track individuals: clustering individual trajectories in crowds using local appearance and frequency trait, in: Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 1467–1474.

[51] A. R. Zamir, A. Dehghan, M. Shah, Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs, in: Computer Vision–ECCV 2012, Springer, 2012, pp. 343–356.

[52] M. Boltes, A. Seyfried, Collecting pedestrian trajectories, Neurocomputing 100 (0) (2013) 127 – 133, ¡ce:title¿Special issue: Behaviours in video¡/ce:title¿. `doi:10.1016/j.neucom.2012.01.036`.
URL `http://www.sciencedirect.com/science/article/pii/S0925231212003189`

[53] S. Baker, I. Matthews, Lucas-kanade 20 years on: A unifying framework, International Journal of Computer Vision 56 (3) (2004) 221–255. `doi:10.1023/B:VISI.0000011205.11775.fd`.
URL `http://dx.doi.org/10.1023/B:VISI.0000011205.11775.fd`

[54] C. Stauffer, Estimating tracking sources and sinks, in: Computer Vision and Pattern Recognition Workshop, 2003. CVPRW'03. Conference on, Vol. 4, IEEE, 2003, pp. 35–35.

[55] M. Nedrich, J. W. Davis, Learning scene entries and exits using coherent motion regions, in: Advances in Visual Computing, Springer, 2010, pp. 120–131.

[56] P. Sand, S. Teller, Particle video: Long-range motion estimation using point trajectories, International Journal of Computer Vision 80 (1) (2008) 72–91.

[57] R. Messing, C. Pal, H. Kautz, Activity recognition using the velocity histories of tracked keypoints, in: Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 104–111.

[58] I. Laptev, On space-time interest points, International Journal of Computer Vision 64 (2-3) (2005) 107–123.

[59] P. Matikainen, M. Hebert, R. Sukthankar, Trajectons: Action recognition through the motion analysis of tracked features, in: Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 514–521.

[60] J. Sun, X. Wu, S. Yan, L.-F. Cheong, T.-S. Chua, J. Li, Hierarchical spatio-temporal context modeling for action recognition, in: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE, 2009, pp. 2004–2011.

[61] J. Sun, Y. Mu, S. Yan, L.-F. Cheong, Activity recognition using dense long-duration trajectories, in: Multimedia and Expo (ICME), 2010 IEEE International Conference on, IEEE, 2010, pp. 322–327.

[62] M. Raptis, S. Soatto, Tracklet descriptors for action modeling and video analysis, in: Computer Vision–ECCV 2010, Springer, 2010, pp. 577–590.

[63] B. Zhou, X. Wang, X. Tang, Random field topic model for semantic region analysis in crowded scenes from tracklets, in: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE, 2011, pp. 3441–3448.

[64] B. Zhou, X. Wang, X. Tang, Understanding collective crowd behaviors: Learning a mixture model of dynamic pedestrian-agents, in: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE, 2012, pp. 2871–2878.

[65] T. Brox, J. Malik, Object segmentation by long term analysis of point trajectories, in: Computer Vision–ECCV 2010, Springer, 2010, pp. 282–295.

[66] W.-C. Lu, Y.-C. Wang, C.-S. Chen, Learning dense optical-flow trajectory patterns for video object extraction, in: Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on, IEEE, 2010, pp. 315–322.

[67] B. Solmaz, B. E. Moore, M. Shah, Identifying behaviors in crowd scenes using stability analysis for dynamical systems, Pattern Analysis and Machine Intelligence, IEEE Transactions on 34 (10) (2012) 2064–2070.

[68] M. Rodriguez, S. Ali, T. Kanade, Tracking in unstructured crowded scenes, in: IEEE 12th International Conference on Computer Vision, ICCV 2009, Kyoto, Japan, September 27 - October 4, 2009, 2009, pp. 1389–1396. `doi:10.1109/ICCV.2009.5459301`.
URL `http://dx.doi.org/10.1109/ICCV.2009.5459301`

[69] S. D. Khan, G. Vizzari, S. Bandini, S. Basalamah, Detecting dominant motion flows and people counting in high density crowds.

[70] A. M. Cheriyadat, R. J. Radke, Detecting dominant motions in dense crowds, Selected Topics in Signal Processing, IEEE Journal of 2 (4) (2008) 568–581.

[71] W. Chongjing, Z. Xu, Z. Yi, L. Yuncai, Analyzing motion patterns in crowded scenes via automatic tracklets clustering, Communications, China 10 (4) (2013) 144–154.

[72] A. Hoogs, A. A. Perera, Video activity recognition in the real world., in: AAAI, 2008, pp. 1551–1554.

[73] S. Khan, G. Vizzari, S. Bandini, S. Basalamah, Detecting dominant motion flows and people counting in high density crowds, JOURNAL OF WSCG 22 (1) (2014) 21–30.

[74] S. D. Khan, G. Vizzari, S. Bandini, Identifying sources and sinks and detecting dominant motion patterns in crowds, Transportation Research Procedia 2 (2014) 195–200.

[75] V. Mahadevan, W. Li, V. Bhalodia, N. Vasconcelos, Anomaly detection in crowded scenes, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE, 2010, pp. 1975–1981.

[76] R. Mehran, A. Oyama, M. Shah, Abnormal crowd behavior detection using social force model, in: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE, 2009, pp. 935–942.

[77] M. Moussaïd, N. Perozo, S. Garnier, D. Helbing, G. Theraulaz, The walking behaviour of pedestrian social groups and its impact on crowd dynamics, PloS one 5 (4) (2010) e10047.

[78] G. Vizzari, L. Manenti, L. Crociani, Adaptive pedestrian behaviour for the preservation of group cohesion, Complex Adaptive Systems Modeling 1 (1) (2013) 1–29.

[79] L. Bazzani, M. Cristani, V. Murino, Decentralized particle filter for joint individual-group tracking, in: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE, 2012, pp. 1886–1893.

[80] M. Zanotto, L. Bazzani, M. Cristani, V. Murino, Online bayesian nonparametrics for group detection, in: Proceedings of British Machine Vision Conference, Surrey. pp, 2012, pp. 111–1.

[81] L. Leal-Taixé, G. Pons-Moll, B. Rosenhahn, Everybody needs somebody: Modeling social and grouping behavior on a linear programming multiple people tracker, in: Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on, IEEE, 2011, pp. 120–127.

[82] J. Sochman, D. C. Hogg, Who knows who-inverting the social force model for finding groups, in: Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on, IEEE, 2011, pp. 830–837.

[83] R. Mazzon, F. Poiesi, A. Cavallaro, Detection and tracking of groups in crowd, in: Advanced Video and Signal Based Surveillance (AVSS), 2013 10th IEEE International Conference on, IEEE, 2013, pp. 202–207.

[84] W. Ge, R. T. Collins, R. B. Ruback, Vision-based analysis of small groups in pedestrian crowds, Pattern Analysis and Machine Intelligence, IEEE Transactions on 34 (5) (2012) 1003–1016.

[85] X. Wang, K. Tieu, E. Grimson, Learning semantic scene models by trajectory analysis, in: Computer Vision–ECCV 2006, Springer, 2006, pp. 110–123.

[86] Z. Fu, W. Hu, T. Tan, Similarity based vehicle trajectory clustering and anomaly detection, in: Image Processing, 2005. ICIP 2005. IEEE International Conference on, Vol. 2, IEEE, 2005, pp. II–602.

[87] I. N. Junejo, O. Javed, M. Shah, Multi feature path modeling for video surveillance, in: Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on, Vol. 2, IEEE, 2004, pp. 716–719.

[88] E. J. Keogh, M. J. Pazzani, Scaling up dynamic time warping for datamining applications, in: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2000, pp. 285–289.

[89] X. Wang, K. T. Ma, G.-W. Ng, W. E. L. Grimson, Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models, International journal of computer vision 95 (3) (2011) 287–312.

[90] E. Grimson, X. Wang, G.-W. Ng, K. T. Ma, Trajectory analysis and semantic region modeling using a nonparametric bayesian model.

[91] T. B. Moeslund, E. Granum, A survey of computer vision-based human motion capture, Computer Vision and Image Understanding 81 (3) (2001) 231–268.

[92] W. Li, X. Wu, K. Matsumoto, H.-A. Zhao, Crowd foreground detection and density estimation based on moment, in: Wavelet Analysis and Pattern Recognition (ICWAPR), 2010 International Conference on, IEEE, 2010, pp. 130–135.

[93] B. D. Lucas, T. Kanade, et al., An iterative image registration technique with an application to stereo vision., in: IJCAI, Vol. 81, 1981, pp. 674–679.

[94] B. K. Horn, B. G. Schunck, Determining optical flow, Artificial intelligence 17 (1) (1981) 185–203.

[95] C. Stauffer, W. E. L. Grimson, Adaptive background mixture models for real-time tracking, in: Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on., Vol. 2, IEEE, 1999.

[96] G. Hamerly, C. Elkan, Alternatives to the k-means algorithm that find better clusterings, in: Proceedings of the eleventh international conference on Information and knowledge management, ACM, 2002, pp. 600–607.

[97] J. Canny, A computational approach to edge detection, Pattern Analysis and Machine Intelligence, IEEE Transactions on (6) (1986) 679–698.

[98] R. Challenger, C. W. Clegg, M. A. Robinson, Understanding crowd behaviours: Supporting evidence, Tech. rep., University of Leeds (2009).

[99] T. Brox, A. Bruhn, N. Papenberg, J. Weickert, High accuracy optical flow estimation based on a theory for warping, in: Computer Vision-ECCV 2004, Springer, 2004, pp. 25–36.

[100] P. Berens, M. J. Velasco, The circular statistics toolbox for matlab, MPI Technical Report No 184.

[101] J.-G. Lee, J. Han, K.-Y. Whang, Trajectory clustering: a partition-and-group framework, in: Proceedings of the 2007 ACM SIGMOD international conference on Management of data, ACM, 2007, pp. 593–604.

[102] J. Sander, M. Ester, H.-P. Kriegel, X. Xu, Density-based clustering in spatial databases: The algorithm gdbscan and its applications, Data mining and knowledge discovery 2 (2) (1998) 169–194.

[103] Z. Zhang, K. Huang, T. Tan, Comparison of similarity measures for trajectory clustering in outdoor surveillance scenes, in: Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, Vol. 3, IEEE, 2006, pp. 1135–1138.

[104] M. Vlachos, G. Kollios, D. Gunopulos, Discovering similar multidimensional trajectories, in: Data Engineering, 2002. Proceedings. 18th International Conference on, IEEE, 2002, pp. 673–684.

[105] A. M. Cheriyadat, R. J. Radke, Automatically determining dominant motions in crowded scenes by clustering partial feature trajectories, in: Distributed Smart Cameras, 2007. ICDSC'07. First ACM/IEEE International Conference on, IEEE, 2007, pp. 52–58.

[106] S. Ali, M. Shah, A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis, in: CVPR, 2007.

[107] M. L. Federici, A. Gorrini, L. Manenti, G. Vizzari, Data collection for modeling and simulation: Case study at the university of milan-bicocca, in: G. C. Sirakoulis, S. Bandini (Eds.), ACRI, Vol. 7495 of Lecture Notes in Computer Science, Springer, 2012, pp. 699–708.

[108] S. Bandini, A. Gorrini, G. Vizzari, Towards an integrated approach to crowd analysis and crowd synthesis: A case study and first results, Pattern Recognition Letters 44 (2014) 16–29.

[109] P. Berens, Circstat: a matlab toolbox for circular statistics, J Stat Softw 31 (10) (2009) 1–21.

[110] J. Shi, C. Tomasi, Good features to track, in: Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on, IEEE, 1994, pp. 593–600.

[111] S. Battiato, G. Gallo, G. Puglisi, S. Scellato, Sift features tracking for video stabilization, in: Image Analysis and Processing, 2007. ICIAP 2007. 14th International Conference on, IEEE, 2007, pp. 825–830.

[112] K. Sankaranarayanan, J. W. Davis, Learning directed intention-driven activities using co-clustering., in: AVSS, 2010, pp. 400–407.

[113] D. T. Campbell, Common fate, similarity, and other indices of the status of aggregates of persons as social entities, Behavioral science 3 (1) (1958) 14–25.

[114] S. Pellegrini, A. Ess, L. Van Gool, Improving data association by joint modeling of pedestrian trajectories and groupings, in: Computer Vision–ECCV 2010, Springer, 2010, pp. 452–465.

[115] F. Solera, S. Calderara, R. Cucchiara, Structured learning for detection of social groups in crowd, in: Advanced Video and Signal Based Surveillance (AVSS), 2013 10th IEEE International Conference on, IEEE, 2013, pp. 7–12.

[116] A. Milan, S. Roth, K. Schindler, Continuous energy minimization for multitarget tracking, IEEE TPAMI 36 (1) (2014) 58–72. `doi:10.1109/TPAMI.2013.103`.