Detection and Simulation of Dangerous Human Crowd Behavior

Dissertation

zur
Erlangung des Doktorgrades (Dr. rer. nat.)
der
Mathematisch-Naturwissenschaftlichen Fakultät
der
Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt
von

Barbara Krausz
aus
Bergisch Gladbach

Bonn, Januar 2012

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn

1. Gutachter: Prof. Dr. Christian Bauckhage

2. Gutachter: Prof. Dr. Sven Behnke

Tag der Promotion: 31.10.2012

Erscheinungsjahr: 2012

Abstract

Tragically, gatherings of large human crowds quite often end in crowd disasters such as the recent catastrophe at the Loveparade 2010. In the past, research on pedestrian and crowd dynamics focused on simulation of pedestrian motion. As of yet, however, there does not exist any automatic system which can detect hazardous situations in crowds, thus helping to prevent these tragic incidents.

In the thesis at hand, we analyze pedestrian behavior in large crowds and observe characteristic motion patterns. Based on our findings, we present a computer vision system that detects unusual events and critical situations from video streams and thus alarms security personnel in order to take necessary actions. We evaluate the system's performance on synthetic, experimental as well as on real-world data. In particular, we show its effectiveness on the surveillance videos recorded at the Loveparade crowd stampede. Since our method is based on optical flow computations, it meets two crucial prerequisites in video surveillance: Firstly, it works in real-time and, secondly, the privacy of the people being monitored is preserved.

In addition to that, we integrate the observed motion patterns into models for simulating pedestrian motion and show that the proposed simulation model produces realistic trajectories. We employ this model to simulate large human crowds and use techniques from computer graphics to render synthetic videos for further evaluation of our automatic video surveillance system.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

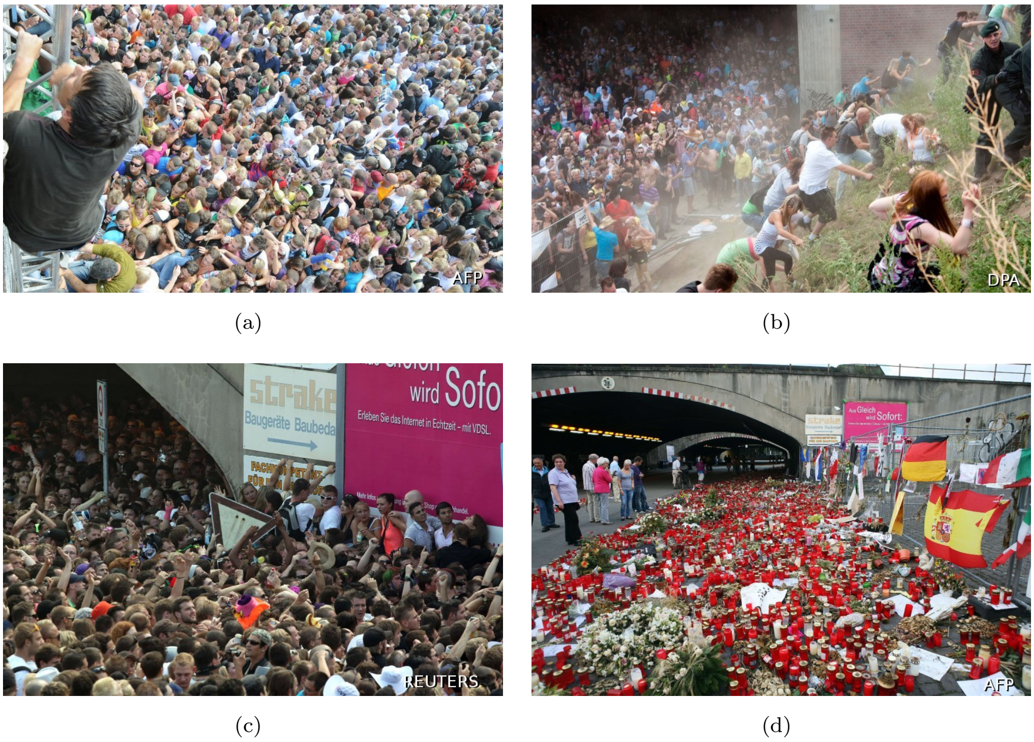
1.1 Motivation

Mass events are and always have been popular in human societies all over the world. Nowadays, typical examples include sports events, festivals, or concerts. Due to increasing populations and higher mobility, mass events attract ever growing numbers of visitors. In addition to an event venue, visitors also use the surrounding facilities in order to reach an event location resulting in high numbers of people in train stations and public spaces.

As an example, the Oktoberfest has more than 6 million visitors with approximately 350 000 visitors each day. Each match of the German football league attracts 40 000 visitors on average where the largest stadium has a capacity of 80 000 people. Train stations such as the main stations in Frankfurt or Hamburg accommodate 350 000 to 450 000 passengers each day. Usually, these large gatherings of people do not cause any problems. But sometimes, despite all precautions, high pedestrian densities might lead to deadly stampedes and crowd disasters.

A recent example of a deadly stampede is the Loveparade crowd disaster that happened at a music festival in July 2010 in Duisburg, Germany. During that event, 21 people died and more than 500 people were injured in a passage way that was too narrow to accommodate thousands of visitors. Figure 1.1 shows press photos of this tragic incident. In Table 1.1, we list examples of other deadly stampedes and crowd disasters showing that human reactions and mass behavior are often unpredictable, in particular, whenever alcohol consumption is an integral part of a mass event.

In crowd disasters, the pedestrian density reaches a critical level causing crowd members to loose their individual control and move involuntarily. In these situations, the crowd



(a)

(b)

(c)

(d)

Figure 1.1: Press photos of the Loveparade crowd disaster. Image credit: AFP, dpa, REUTERS.

pressure increases making it difficult to breathe. Also, shock waves propagate through the crowd and cause people to stumble and to fall down. In crowd stampedes, most people die by suffocating due to high pressure on their chests. According to Fruin [4], forces can reach 4500 N/m and are caused by horizontal pushing of the crowd as well as people being stacked up vertically. Studies report that the stack of bodies was 3m high at some crowd disasters [4].

Typically, high pedestrian densities occur in locations with limited capacity such as bottlenecks. However, high densities might also be caused by the crowd itself. In a "craze", people rush in order to reach a goal, for instance, an optimal position near the entertainer in a concert. In contrast, crowd stampedes are also often caused by people escaping from a site due to real or perceived danger.

A major problem in large crowds is the lack of communication. While being part of a crowd, individuals can only communicate with their local neighborhood and have a limited view of the whole crowd. This leads to a false perception of movement when people are collapsing and people in the rear perceive a movement and move forward thereby increasing the pressure even more.

The frequency of these incidents as well as the number of casualties illustrate the need

Year	Place	Deaths
2011	Stadium, Bamako (Mali)	>36
2011	Campaign Rally, Port Harcourt (Nigeria)	11
2011	Nightclub, Budapest (Hungary)	3
2011	Pilgrimage, Kerala (India)	102
2010	Water Festival, Phnom Phen (Cambodia)	>375
2010	Loveparade, Duisburg (Germany)	21
2006	Stadium, Ibb (Yemen)	51
2006	Pilgrimage, Mina (Saudi Arabia)	363
2005	Religious Procession, Baghdad (Iraq)	>640
1999	Subway Station, Minsk (Belarus)	53
1990	Pilgrimage, Mina (Saudi Arabia)	1426
1989	Hillsborough Stadium, Sheffield (UK)	96
1982	Stadium, Moscow (Russia)	340

Table 1.1: Examples of deadly stampedes and crowd disasters (see [2, 3]).

for further research in the area of crowd management and crowd control. Modern technologies for crowd management include video surveillance for monitoring large areas and buildings. Here, a large number of cameras is installed and the video streams are relayed to a control room as shown in Figure 1.2. Typically, human operators observe these displays where they have to view multiple displays and control cameras in parallel. However, studies indicate that the attention span of a human operator observing screens of a surveillance system ranges between thirty minutes and two hours and that most operators can monitor up to four displays in parallel [5]. Moreover, in large facilities, typically, the number of cameras is much higher than the number of video monitors. For instance, in Turin, Italy, more than 800 cameras are installed in the metro, but only 24 displays show the video streams in a control room [6] leading to a vast amount of video streams that is never being watched by human operators.

In cases like these, automatic video surveillance systems may assist human operators. At a mass event, for instance, a video surveillance system may help to estimate the visitor density and to detect indicators of critical situations. This can buy crucial time in which security personnel can be dispatched and streams of pedestrians can be redirected.

However, the application domains of automatic video surveillance systems are highly demanding. First of all, such a system must work in real-time and, ideally, it needs as few interactions with human operators as possible, e.g. for calibration purposes. Next,

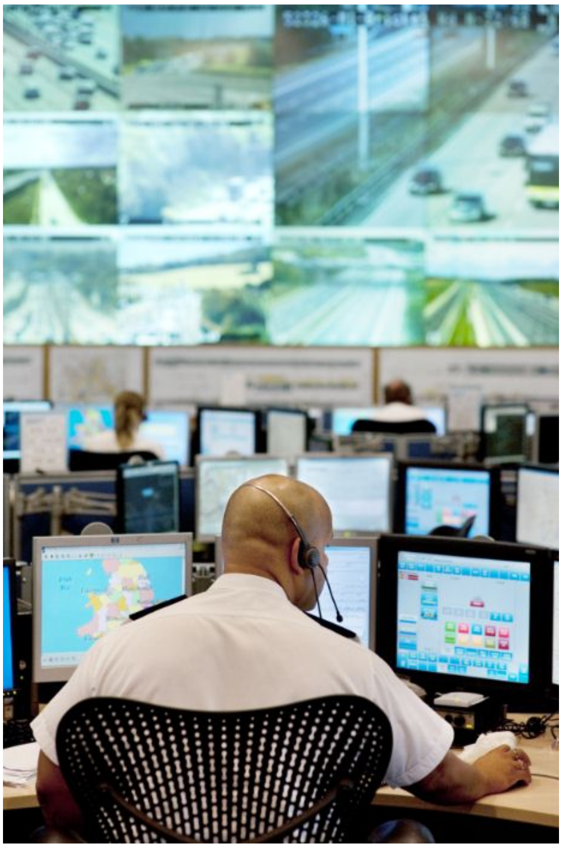


FIGURE 1.2: Control room with human operators watching multiple video streams. Image credit: Highways Agency.

the false alarm rate of the system should be low which is even more important for large camera networks. In particular, when observing humans, the privacy of the people being monitored has to be preserved.

In addition to that, automatic analysis of video streams is a highly challenging research topic. Amongst others, changing illumination conditions, the variability of people's appearances and actions as well as the lack of training data are common problems an automatic surveillance system has to cope with. In the particular case of large human crowds, we are faced with thousands of people visible in the scene leading to severe occlusions, thus making conventional tracking approaches unfeasible.

1.2 Contributions

In this work, we study pedestrian behavior in human crowds. In particular, we investigate crowd behavior at the Loveparade stampede from a computer vision point of view.

Our main contribution is the observation and quantitative study of characteristic motion patterns in congested areas as well as the development of an automatic computer vision system that makes use of these observations and is able to detect unusual events and critical situations such as congestions in crowds. Secondly, we improve physical models for simulating pedestrian motion by integrating the observed characteristic motion behavior in dense crowds and show that the improved model indeed simulates realistic trajectories and reproduces common self-organizing phenomena. Thirdly, we involve pedestrian simulation and computer graphics in order to synthesize videos of human crowds which are then used to evaluate our proposed computer vision techniques.

1.3 Outline

First, we give an overview of works related to crowd dynamics in Chapter 2. In particular, we present works in pedestrian simulation, computer vision and empirical studies of pedestrian motion as well as works that are positioned in the intersection of these research areas. In Section 2.2, we investigate pedestrian behavior in congestions on a microscopic level as well as pedestrian behavior in dense crowds in Section 2.3.

Chapter 3 first reviews the crowd incident at the Loveparade. Then, we conduct a series of experiments on the video data and recognize the motion pattern that we discussed in Chapter 2. Next, we develop a system for automatic detection of unusual events and congestions in particular.

In Chapter 4, we evaluate the developed system on synthetic, experimental as well as on real-world datasets and show its effectiveness.

Chapter 5 covers pedestrian simulation. First, we discuss a physical model for simulating pedestrian motion. Then, we improve this model by integrating the characteristic motion behavior that we observed in Chapters 2 and 3 and evaluate the proposed model. Finally, we simulate pedestrian motion and use techniques from computer graphics to render videos of large human crowds. We use these artificial videos to further evaluate our computer vision system developed in the thesis at hand.

Finally, Chapter 6 draws a conclusion and gives an outlook.

Chapter 2

Pedestrian Behavior in Crowds

Whenever the number of pedestrians exceeds the capacity, congestions occur in which unobstructed pedestrian flow becomes impossible. In contrast to vehicular traffic [7], congestions are not only annoying, but might become very dangerous. For example, when an event location has to be evacuated, people might be trapped in a congestion and cannot escape early enough.

In other cases, the density of pedestrians might become extremely high; studies report densities up to 10 people per square meter [8]. In such densely packed crowds, people moving involuntarily induce sudden movements of other people nearby. These *crowd turbulences* propagate through the crowd causing people to stumble and to fall down. In these situations, people might suffocate due to enormous pressure.

Work towards the prevention of crowd disasters is often based on experiments and the analysis of video footage from crowd catastrophes. Researchers try to understand the mindset of panicked people and develop physical models in order to simulate human behavior in crowded environments, in particular to devise evacuation strategies. Using such simulations, one can identify and defuse places in an environment that are potentially dangerous. For instance, a study by Helbing et al. [9] of video footage from a crowd disaster in Mina during the annual Muslim pilgrimage led to structural alteration of the site and helped to re-organize the pilgrimage. Indeed, since then, there has not been another deadly stampede during the Hajj.

In Section 2.1, we review works related to crowd dynamics. In particular, we briefly present different models for pedestrian simulations, give an overview of empirical studies as well as computer vision approaches designed to work in dense crowds. Next, we analyze human motion behavior in congested areas on a microscopic level. Finally, pedestrian behavior in densely packed crowds is covered in Section 2.3.

2.1 State of the Art in Crowd Dynamics

Pedestrian dynamics have been studied intensively for more than 40 years. Knowledge about pedestrian dynamics and human behavior in crowds is of great interest in many applications [10]: In *crowd management*, strategies for preventing accidents and crowd disasters are developed. During an event, crowd managers observe the crowd and identify potentially dangerous locations. In *public space design*, pedestrian behavior is analyzed in order to support planning of urban infrastructure, for instance, in the design of pedestrian facilities. Here, an important aspect is the architectural design of buildings and public spaces for improving evacuation strategies. Physical models of pedestrian behavior in combination with *virtual environments* can be used for simulating crowd phenomena which are of great importance in computer games and movies, for instance. Recently, in *visual surveillance*, methods for observing crowded scenes and detecting dangerous situations are being developed for supporting security personnel. Lastly, knowledge about human crowd behavior is used in *intelligent environments* in order to control and direct streams of pedestrians.

Researchers from various areas contribute to pedestrian dynamics and crowd behavior understanding. Among them are physicists and engineers who develop physical models for pedestrian simulation and evacuation simulations in particular. Sociologists and psychologists aim at understanding the mindset of people whereas computer scientists develop computer vision methods for automatically analyzing video footage and detect anomalies and raise alarms.

In Sections 2.1.1 to 2.1.3, we present related work originating from the above mentioned research areas. Section 2.1.4 discusses works that are positioned in the intersection between different research areas.

2.1.1 Physical Models for Pedestrian Simulation

In order to prevent congestions and overcrowding and to improve evacuation strategies in emergency situations, human motion behavior is analyzed and, based on these findings, pedestrian models for simulations are developed. Human motion behavior is studied at different levels of abstraction: At the *strategic level*, humans choose their activities, such as "go to the supermarket". The concrete route choice for reaching these goals and other short-term decisions are made at the *tactical level* while taking obstacles, walls and other pedestrians into account. The actual movement of a human is described at the *operational level* with respect to characteristics of human motion behavior. Mostly, pedestrian models for simulations are working on the operational level.

They can be categorized into continuous as well as discrete models. Further, physical models of pedestrian behavior can be distinguished into *microscopic*, *macroscopic* and *mesoscopic* models based on their level of abstraction [11]. *Microscopic* models [12–14] model the behavior of each individual pedestrian as well as interactions between individuals. Although they model human behavior quite realistic and have been shown to reproduce self-organizing effects, microscopic models are computationally demanding. *Macroscopic* models [15–17] treat a crowd as a whole and describe pedestrian dynamics in terms of average flow and velocity, whereas *mesoscopic* models [18, 19] adopt ideas from both approaches by grouping individuals and considering the crowd as the sum of groups with different behavior.

Fluid-dynamic models [15–17] are the most widely-used macroscopic pedestrian models. Taking similarities between dense crowds of pedestrians and fluids into account, Hughes [16] models a crowd as a continuous density field and describes the evolution of crowd dynamics by equations adopted from fluid dynamics.

In contrast to macroscopic models, microscopic models consider individual movements of pedestrians. An early discrete microscopic model is proposed by Løvås [14]. In his *queueing model*, a building is modeled as a graph with rooms as nodes. Two nodes are connected by an edge, if there exists a door between both rooms. Pedestrians are modeled as queueing network customers with individual attributes such as walking speed and a destination that a pedestrian wants to reach. A number of pedestrians is fed into the network and individual movements are simulated based on route choice models.

Cellular automata [13, 20] are discrete models both in space and time. The environment is discretized into cells which can be occupied by at most one pedestrian. In discrete time steps, each pedestrian is moved to an unoccupied neighboring cell based on simple rules. In [13], Burstedde et al. additionally superimpose static and dynamic force fields that alter transition probabilities of pedestrians moving on the grid. These fields model obstacles and regions being more attractive to pedestrians as well as influences of other pedestrians.

Both, queueing models and cellular automata, are discrete in space and time and therefore have difficulties in reproducing typical phenomena observed in human crowds (see Sections 2.2 and 2.3). The most popular continuous microscopic model is the *social force model* proposed by Helbing et al. [12, 21] which assumes socio-psychological and physical forces that influence human behavior. Here, each pedestrian has a preferred walking speed and walks towards a destination that he wants to reach. The environment including walls and obstacles exerts forces on pedestrians whereas socio-psychological forces model repulsive forces between individuals. A pedestrian moves into the direction of his destination while being influenced by these forces. The social force model has been studied intensively in recent years and many modifications [22–28] have been proposed to improve the accuracy of the model and to reproduce self-organizing effects in human crowds.

2.1.2 Empirical Studies for Understanding Human Behavior

Simulations based on models for pedestrian dynamics as presented in Section 2.1.1 reflect human motion behavior in different environments and situations. In order to make simulations more realistic and to validate simulation results, empirical data is of great importance. Biomechanical and physiological properties of humans as reported by Weidmann [29] or Liu et al. [30], for instance, are important variables in simulations. Examples of such properties include body proportions, space requirements, walking speeds and other walking characteristics.

In particular, effects on human motion behavior caused by the environment as well as other pedestrians have been studied and form the basis of pedestrian models. Ideally, empirical data is obtained from real data. However, since it is rather difficult to obtain real data annotated with ground truth because of complexity as well as privacy concerns, researchers conduct experimental studies and determine parameters such as crowd density, speed, flow, and crowd pressure either manually [24] or by means of digital image processing [30–32].

In most experimental studies, the *fundamental diagram*, the relationship between velocity and density, is studied under different conditions [24, 29]. This relationship is of great importance, since it quantifies the capacity of facilities and gives an estimate of evacuation strategies. In general, the velocity decreases with increasing density. However, experiments study the influence of different environments like stairs, corridors and bottlenecks as well as variations due to pushy movements of pedestrians and cultural differences [33, 34].

Human motion behavior and typical self-organizing effects are studied in different environments. Kretz et al. [20], for example, conduct experiments with two pedestrian groups with contrary moving directions in a corridor. They study lane formation as well as walking speeds and flow for varying group sizes. The walking speeds of pedestrians on stairs is analyzed by Fujiyama and Tyler [35]. In their experiments, they test the influence of different physical characteristics (weight, age etc.) and characteristics of the stairs on the walking speeds. Since motion behavior and clogging effects at bottlenecks is of major interest in evacuation situations, many experiments have been carried out that examine the influence of the shape and the width of bottlenecks on the walking speeds and pedestrian flow [24, 36–39]. Other experimental studies [40, 41] concentrate on congested pedestrian traffic and find crowd phenomena such as stop-and-go waves occurring in congested areas.

In addition to experiments conducted under laboratory conditions, some researchers observe evacuation exercises. Ashe and Shields [42] evacuate a retail store and compare evacuation times to simulated evacuation times. Similarly, Weckman et al. [43] assess different simulation models using data obtained from an evacuation in a theater, whereas Klüpfel et al. [44] compare simulation results to an evacuation exercise in a primary school.

Recently, phenomena in dense crowds such as stop-and-go waves and crowd turbulences (see Sections 2.2 and 2.3) have been discovered [9]. Stop-and-go waves are indicators of critical pedestrian densities. In even more densely packed crowds, crowd turbulences occur and shock waves propagate through the crowd causing people to stumble, fall down and even suffocate. Due to its nature, there is a lack of data showing such critical motion patterns. In fact, the empirical study of Helbing and Johansson [9, 31] of a stampede during the Muslim pilgrimage with 363 casualties is the only study of real video data of a crowd disaster. It gives valuable insights into crowd dynamics during critical situations and leads to important extensions of pedestrian models [28]. However, the findings of Helbing and Johansson have not yet been investigated by computer vision researchers. Thus, to the best of our knowledge, there does not exist any system for detecting critical crowd conditions in real-time and for preventing crowd disasters.

2.1.3 Computer Vision Techniques for Crowd Analysis

The analysis of human motion has been and is still an active research area in computer vision. Traditionally, objects of interests are detected and tracked throughout a video sequence. [45], [46] and [47] give an extensive overview of techniques for detection and tracking objects.

However, conventional approaches are not feasible in crowded scenes due to inappropriate camera viewpoints and severe occlusions. Techniques specifically designed to work in scenes with occlusions can be found for example in [48] where Mikolajczyk et al. use part-based detectors for human detection or in [49] where local and global cues are combined to detect partially occluded people. Although these methods work in real-world scenes with severe occlusions, they are not appropriate in dense crowds with hundreds and thousands of humans. Moreover, tracking individuals necessitates considerable computational effort making real-time applications difficult or even impossible.

2.1.3.1 Crowd Density Estimation

Recently, holistic approaches have been proposed that treat a crowd as a whole instead of detecting and tracking individuals. In contrast to traditional object-based methods that extract trajectories for individual people, holistic approaches compute global motion features, such as spatio-temporal gradients or optical flow. A majority of holistic approaches deals with automatic density estimation of a crowd. The density of a crowd is a valuable piece of information especially in crowd management. A crowd density increasing to a very high level might lead to dangerous stampedes. Having detected high densities, streams of pedestrians can be redirected. The estimated number of people is also very interesting in public transport management, for example, as the transport manager can react to an increasing passenger quantity by adapting the resource planning.

Most of these works apply background subtraction techniques and learn a relationship between foreground pixels, blob features, or edge features and the crowd density. For example, Davies et al. [50] apply background subtraction and edge detection and relate values obtained from these features to the number of people visible in the scene. Interestingly, their system distinguishes moving from static crowds using the fact that a moving crowd shows up-and-down oscillations of the heads. For obtaining an estimation of the frequency of the head oscillations, they exploit frequency-domain techniques.

In addition to background subtraction, some approaches exploit texture features for estimating the crowd density. Lo and Velastin [51] combine features obtained from background subtraction with texture features as inputs to a multi-layer perceptron to classify the density in a subway station into four categories. Chan et al. [52] make use of the gray-level co-occurrence matrix which denotes the probability of a pair of gray-levels co-occurring in a defined neighborhood. Given the gray-level co-occurrence matrix, they infer texture features such as homogeneity, energy and entropy. A comparison of density estimation results obtained using different texture features can be found in [53].

2.1.3.2 Detection of Unusual Events and Emergency Situations

The huge amount of surveillance cameras in public spaces comes along with an increasing need to automatically analyze video data and especially to detect anomalies and emergency situations. Typically, two approaches for event detection can be distinguished: *Explicit event detection* methods and *deviation methods.*

Explicit event detection methods model events that the system should identify [54–57]. Typically, these events are threatening or suspicious events like people fighting, loitering, cars driving into the wrong direction, etc. An example for an explicit event detection method can be found in [54]. Here, Boghossian and Velastin detect crowd-related emergencies like overcrowding, bottlenecks, and obstacles by identifying and segmenting different motion directions in crowds. Locations with circular flow paths are detected in a Hough voting scheme and may be indicative for bottlenecks at scene exits. Similarly, locations with crowd flow pointing away from it are detected and indicate potential dangers like fire or fights, for instance. They also identify obstacles which are regions without any distinct motion vectors. Their system assists a human observer at mass events by superimposing identified regions of consistent motion vectors as well as potential dangers, exits and obstacles.

Explicit event detection methods need to model each possible event which is mostly unfeasible in real-world scenes due to the number of possible events and the lack of training data showing suspicious events. In contrast, *deviation methods* learn models of usual activities or dominant routes and detect deviations from the learned models to identify anomalies. Usually, enough training data showing usual events is available making this approach applicable in real-world settings in contrast to event detection methods that need training data for modeling suspicious events.

Approaches for unusual event detection can further be distinguished by the type of feature: The majority uses sparse [58–60] or dense [61–63] optical flow, whereas other approaches make use of spatio-temporal gradients [64] or spatial gradients without any temporal information [65].

A simple and fast method for abnormal event detection based on sparse optical flow is proposed by Adam et al. [59]. They compute optical flow and derive magnitude or direction at a fixed set of spatial positions. By taking into account the history of magnitude or direction values at that position, a histogram of typical values can be constructed. Given this histogram, the likelihood of a new observation can be computed. An unusual event is detected, if the likelihood values at multiple spatial positions are low. Adaptation to a changing environment is achieved by deleting old observations and updating the histogram with new observations.

Andrade et al. [66] compute optical flow fields and use spectral clustering for determining a set of normal motion patterns. For each set, they train a Multiple Observation Hidden Markov Model (MOHMM). A new observation is classified as a normal or as an unusual motion pattern based on its likelihood computed given the bank of Hidden Markov Models.

Wang et al. [58] propose an approach for anomaly detection based on optical flow and Hierarchical Bayesian Networks. As a low-level visual feature, they compute motion direction of moving pixels which are obtained by frame differencing. The frame is divided into cells and each moving pixel is assigned a word of a codebook based on its motion direction and rough position in the frame. A video sequence is divided temporally into clips which are considered as documents. Using an extension of Hierarchical Dirichlet Process (HDP), topics of visual words can be discovered providing a set of typical atomic activities in a scene. Their framework is capable of discovering co-occurring topics which are explained as typical interactions in a scene. Abnormal events or even abnormal interactions are detected by identifying low likelihood values based on the learned Bayesian models.

Kim and Grauman [63] make use of dense optical flow as a low-level feature for abnormal activity. They detect irregularities locally, but combine features globally in order to detect anomalies at a higher scale. Thereto, they compute a dense optical flow field at different scales and subsume optical flow values to obtain a high-dimensional optical flow feature vector for each sub-region of the video frame. A generative model for local activity patterns is learned by computing a common Mixture of Probabilistic Principal Component Analyzers (MPPCA) for all sub-regions. Then, for each sub-region, they construct a frequency histogram that counts how often an MPPCA component is observed. Additionally, a co-occurrence histogram denotes the frequency of two components co-occurring at adjacent sub-regions. In order to incorporate spatiotemporal information, a Markov Random Field (MRF) is constructed in which each node corresponds to a sub-region. They define local potential functions and pair-wise potentials using the frequency and co-occurrence histograms. Finally, for a test video, the nodes of the MRF are labeled as normal or abnormal using maximum a posteriori estimation. In order to account for dynamically changing environments, their model can be updated incrementally by updating the histograms and the MPPCA parameters.

Kratz and Nishino [64] divide the video into fixed size cuboids and determine a 3D Gaussian distribution of spatio-temporal gradients within each cuboid. Cuboids that exhibit similar motion patterns are identified in a clustering procedure, thus providing prototypes of typical motion patterns for each spatial location. Unusual events are simply detected by identifying motion patterns with low likelihood values given the prototypes of motion patterns in that location. In order to capture temporal information, Hidden Markov Models are learned for each spatial location using the extracted prototypes of motion-patterns as observations. Spatial relationship between locations is taken into account by constructing a coupled Hidden Markov Model covering four neighboring locations. Combining temporal likelihood and spatial likelihood linearly, yields a confidence measure for unusual motion patterns.

Mahadevan et al. [67] propose to use dynamic textures as low-level features. The basic representation used herein is a mixture of dynamic textures which capture dynamics and appearance. The observed video sequence is considered as a sample from one of K dynamic textures. In a training phase, a mixture of dynamic textures (MDT) has been learned for each region of the scene. Similar to the well-known background subtraction approach based on pixel-wise Gaussian Mixture Models (GMM), the learned MDTs are used for computing the probability of the spatio-temporal patches. For detecting spatial anomalies, discriminant saliency based on mixtures of dynamic textures is used. Salient regions are those regions that exhibit different patterns compared to their surrounding.

Recently, approaches based on *particle advection* have been proposed [60, 68–71]. Here, particles are placed onto a grid and moved according to the underlying optical flow field. Ali and Shah [68] segment the resulting flow field to identify regions of homogeneous flow. Instabilities of the flow are detected when the number of homogeneous flow segments change. In [69], Ali and Shah extended this approach to track individuals in crowded scenes. They use their previous technique to identify obstacles and dominant paths and integrate this information into a probabilistic tracking framework.

Mehran et al. [60] combine the approach of particle advection with the social force model [12]. The velocity of the particles is integrated into the social force model which models interaction forces between particles. Interaction forces are estimated for a video segment and spatio-temporal visual words are clustered to obtain a codebook. Treating video segments as documents, different topics of normal crowd behavior are extracted by means of Latent Dirichlet Allocation. For labeling frames as normal or abnormal, the likelihood for a video segment is computed given the learned model.

In [70], Widhalm and Brändle apply particle advection to identify major routes of pedestrians and to detect abnormal particle trajectories. Points of obtained particle trajectories represented by spatial and motion information are vector quantized in order to obtain a codebook of prototypes. Particle trajectories can now be represented as sequences of prototypical motion events. For obtaining dominant routes of people visible in the scene, particle trajectories are clustered using spectral clustering. The pairwise distances between trajectories are obtained by using Dynamic Time Warping. Having extracted dominant routes, new particle trajectories can be assigned to one of the clusters. An abnormal trajectory is detected, if it cannot be assigned to a cluster.

2.1.4 Bridging the Gap between Computer Vision and Crowd Dynamics

Extensive research has been done in crowd dynamics, computer vision, crowd synthesis, and computer graphics. Although these research areas are closely related, it is remarkable, that there exists little work that adopts ideas from multiple research areas. In fact, various scenarios connecting different research areas can be identified: Firstly, computer vision techniques can help to get insights into the behavior of pedestrians for improving pedestrian models. Mainly, these techniques reduce manual labor when extracting trajectories or quantities like velocity or density. On the other hand, physical models for human behavior can guide computer vision algorithms. Motivated by the idea, that humans exhibit particular motion characteristics in crowds, this information can be incorporated into visual tracking techniques or abnormal event detection methods. Next, crowd simulations may be used in combination with computer graphics to synthesize artificial video which can then be used in validation or training of computer vision systems. Finally, one can also use computer vision methods to extract information from real-world videos in order to enhance crowd simulations making them more realistic. In the following, we present some works situated in the intersection between these research areas.

Researchers working in the area of pedestrian dynamics apply visual tracking techniques to extract trajectories of individuals which help to validate and improve physical models for pedestrian dynamics. For example, in an experimental setup, Liu et al. [30] detect and track people using mean-shift tracking to obtain velocities and other trajectory features. Similarly, Boltes et al. [32] apply computer vision techniques on video recordings of experiments in order to obtain pedestrian trajectories which are highly accurate in space and time. Johansson and Helbing [31] analyze real-world videos showing pilgrim flows in Mina, Saudi-Arabia. They employ digital image processing for minimizing manual labor in order detect people and to determine local densities, speeds and pedestrian flows. Other approaches that use automatic video analysis can be found in [72, 73], for example. However, these works typically use experimental data or analyze non-crowded real-world scenes, mostly recorded from a topview camera.

Recently, computer vision researchers have integrated physical models of pedestrian flows in order to facilitate visual tracking. The first work that integrates pedestrian behavior into a tracking framework is proposed by Antonini et al. [74]. They introduce discrete choice models for pedestrian behavior which are trained on real video sequences. Ali and Shah [69] present a tracking framework inspired by the cellular automaton model [13]. They automatically calculate force fields that integrate information on human behavior as well as the locations of obstacles and important regions such as exit doors.

This information is integrated into their probabilistic tracking framework. Pellegrini et al. [75, 76] replace a standard first-order motion model that predicts a pedestrian's position by a motion model that takes social behavior into account and is inspired by the social force model. Similarly, Mehran et al. [60] adopt ideas from the social force model [12] and estimate interaction forces between particles in order to detect abnormal events.

A major problem in the development of computer vision algorithms for crowd-related problems, such as crowd density estimation or unusual event detection is the lack of real-world data. Hence, some researchers use computer graphics to generate artificial videos which are then used for validation and training of computer vision algorithms. For instance, Wu et al. [55] generate synthetic images of varying crowd density in order to train a support vector machine. In [77], Andrade and Fisher make use of the social force model [12] and simulate pedestrian behavior. They render artifical scenarios showing abnormal behavior which is detected by their computer vision framework for unusual event detection proposed in [66]. Similar, Allain et al. [78] use the social force model to simulate virtual characters. Additionally, they enhance their simulated videos with walking motion obtained from motion capture data. The resulting synthetic videos are analyzed by their vision algorithm in order to determine the velocity field in the scene as well as the disturbance potential, a measure which is related to the density and the pressure in the crowd.

Works related to human crowd synthesis make use of computer vision techniques in order to render realistic animations of human crowds. In contrast to the works of Andrade and Fisher [77] as well as Allain et al. [78], they do not use a physical model for pedestrian dynamics as the primary input for their synthesized animations. Instead, they analyze video footage showing human crowds for guiding the motion of virtual agents and, finally, render a synthesized animation. Courty and Corpetti [79], for example, compute optical flow fields from real-world videos of human crowds and map these motion fields from the image domain to the virtual world domain. Virtual characters are placed into the virtual world and moved according to the computed motion field. The synthesized animations are shown to reproduce real-world scenes with reasonable accuracy. Similarly, Musse et al. [80] track humans in non-crowded real-world videos and cluster the obtained trajectories. Using the cluster centers they are able to simulate virtual pedestrians exhibiting motion characteristics similar to the tracked humans. When simulating dense crowds, Musse et al. also take into account the social force model [12] so that simulated agents in dense crowds will mainly follow the learned prototypical trajectories while at the same time avoiding collisions by computing interaction forces.

2.2 Pedestrian Behavior in Congestions

2.2.1 The Fundamental Diagram

According to Schadschneider et al. [81], congestions occur in locations of high density where the inflow exceeds the capacity. These locations are called bottlenecks and have been studied for example in [37–39]. Besides bottlenecks, jamming can also be observed in locations where two opposing pedestrian streams clash resulting in high densities and low velocities.

In pedestrian dynamics, the relationship between pedestrian density ρ and velocity v is captured by the *fundamental diagram.* The most frequently cited version is the fundamental diagram given by Weidmann [29] who evaluated 25 different datasets. Basically, the fundamental diagram shows decreasing velocities for increasing densities, although diagrams found by others [9, 82, 83] differ from the graph shown in Figure 2.1.

Seyfried et al. [24] sub-divide the range of densities into four different domains each having a different decrease in velocity. Essentially, the domains III ($2.3^{-2} \leq \rho \leq 4.7^{-2}$) and IV ($\rho \geq 4.7^{-2}$) (see Figure 2.1) correspond to congestions where physical contact between pedestrians is hardly avoidable. Here, the velocity has dropped to a low level (< 0.5 m/s), whereas for normal walking conditions, the velocity is around 1.34 m/s on average [29].

In their later work [41], Seyfried et al. investigate the relationship between density and velocity on a microscopic level by tracking single pedestrians and measuring microscopic motion characteristics based on obtained trajectories [84]. Observing the probability distribution of velocities for fixed densities, they find two peaks for high densities reflecting the two phases of *stop-and-go waves.* Stop-and-go waves show alternating forward pedestrian motion and backward gap propagation and have been observed by researchers in locations of high pedestrian density where an unobstructed pedestrian flow becomes impossible [9, 85, 86].

2.2.2 Pedestrian Behavior in Congested Areas

On a microscopic level, pedestrian behavior at various velocities give valuable insights into human behavior in congested areas. We analyze trajectories obtained from video recordings of a large scale experiment conducted under laboratory conditions [40, 87]. Pedestrian groups of varying sizes ($N = 14, 17, \ldots, 62, 70$) walk through a corridor recorded by topview cameras.

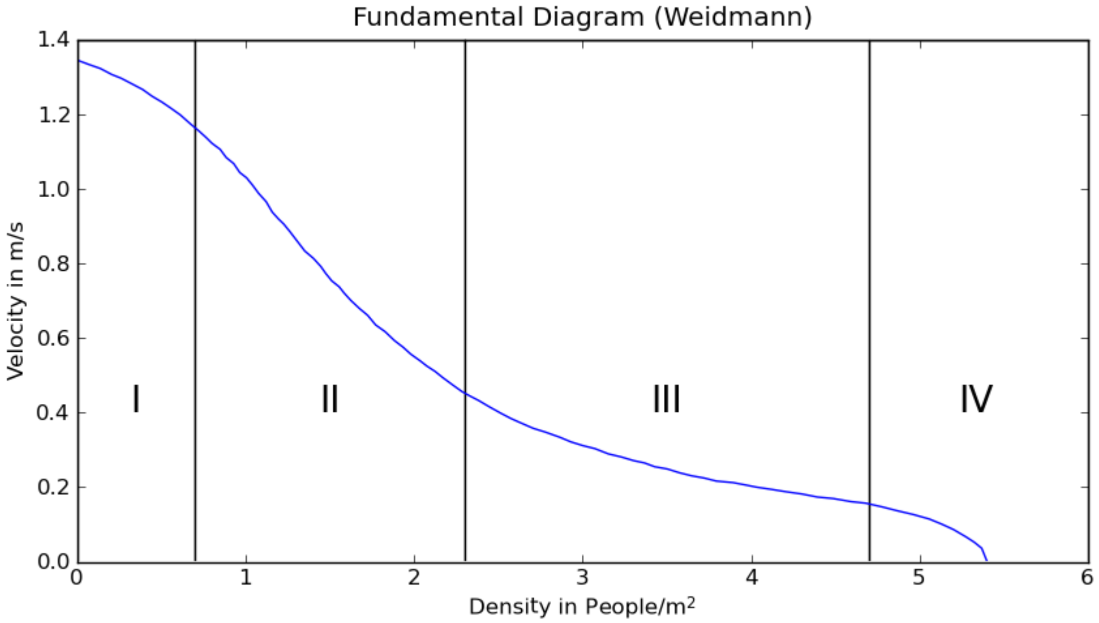


Figure 2.1: Fundamental diagram according to Weidmann [29] measuring the relationship between density ρ and velocity v. The diagram is based on 25 different datasets. The range of densities is sub-divided into four domains of different decrease of velocities where domains III and IV correspond to congestions, see [24].

As expected, we first observe that the mean velocity of the pedestrians decreases for increasing group sizes, see Figure 2.3. Note that at the end of some experiments (for $N = 56, 62, 70$), the velocities increase due to the settings of the experiment resulting in higher mean velocities.

Second, from the trajectories, we can observe that people do not move along a straight line, instead, it is a characteristic of human gait, that humans tend to swing laterally. Furthermore, comparing trajectories of pedestrians walking at different speeds reveals substantial differences. One can clearly see differences both in the frequency and the amplitude of lateral swaying. Similar findings have been reported by Hoogendoorn and Daamen [37] and Liu et al. [30].

Figure 2.4 shows the relationship between mean velocities and mean amplitudes of lateral oscillation. We find amplitudes in the range between 6 cm and approx. 30 cm. Fitting a linear model, we find a negative relationship between the amplitude a (in m) and the velocity v (in m/s):

$$a = -0.14v + 0.21 \tag{2.1}$$

which is comparable to the findings of Liu et al. [30] as well as Hoogendoorn and Daamen [37]. We argue that humans tend to swing laterally when walking slowly in order to keep their balance.

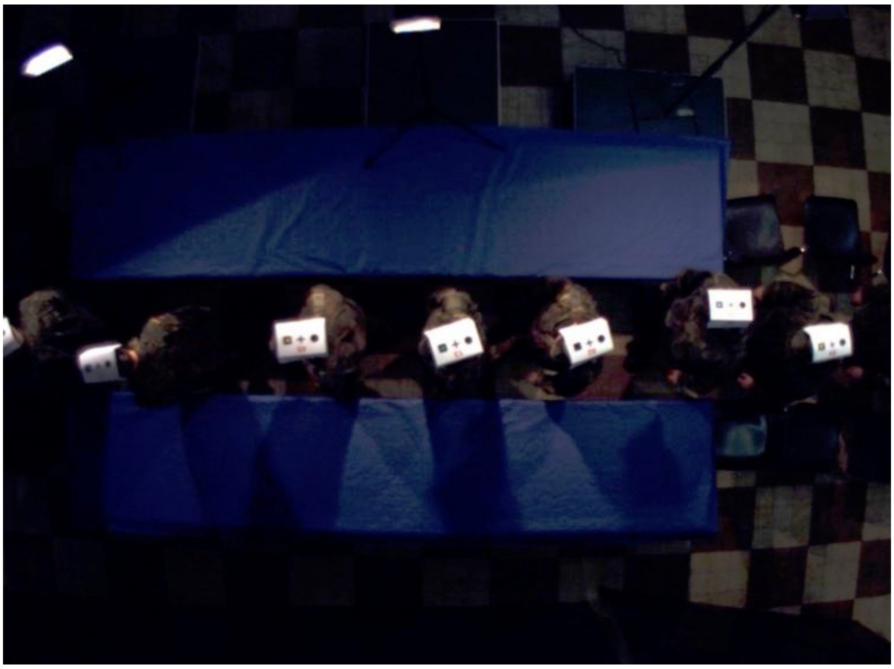


Figure 2.2: Experiment with pedestrian groups of varying sizes recorded from topview cameras [40, 87].

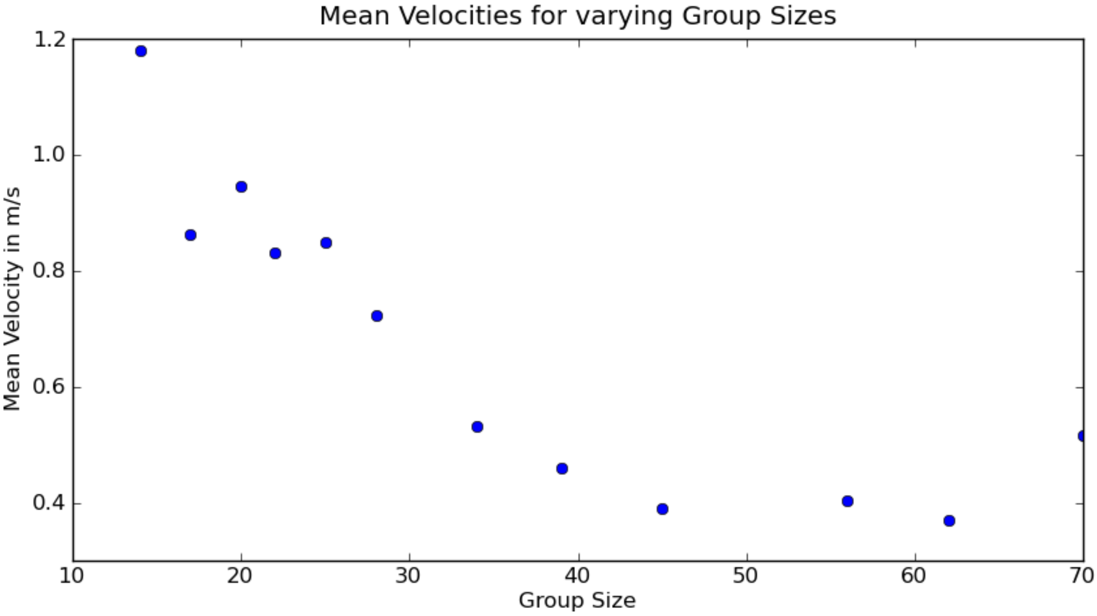


Figure 2.3: Mean velocities depending on group size. Note that due to the settings of the experiment, the velocities increase at the end of the experiment resulting in high mean velocities for $N = 56, 62, 70$.

Next, we analyze the frequency of lateral oscillations which is depicted in Figure 2.5. Similar to amplitudes, we find a linear relationship between frequencies and velocities, but, in contrast, frequencies increase with increasing velocities. By fitting a linear model,

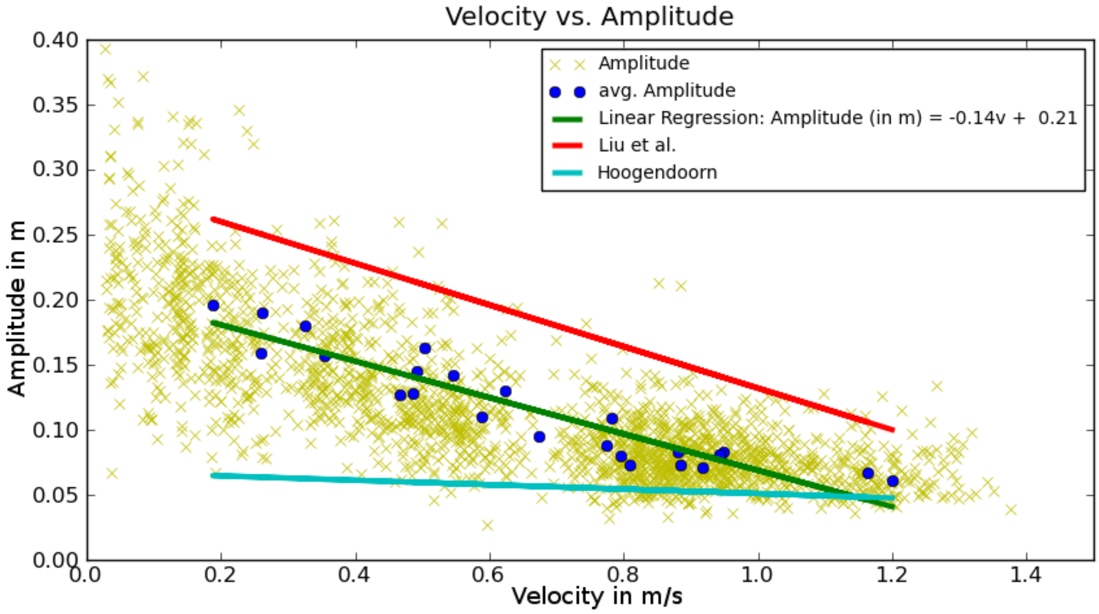


FIGURE 2.4: Relationship between velocity and amplitude of lateral oscillation in comparison to the findings of Liu et al. [30] as well as Hoogendoorn and Daamen [37]. Blue markers correspond to mean velocity and mean amplitude of different pedestrian groups.

we find that the frequency f (in Hz) is related to the velocity v (in m/s) as follows:

$$f = 0.44v + 0.35. \tag{2.2}$$

Although Liu et al. also find a linear relationship, the frequencies observed in their experiments are higher than in our experiments. Hoogendoorn and Daamen fit a quadratic function which is in accordance to the results reported in [30], see Figure 2.5.

Having analyzed space requirements in lateral direction, we also analyze space requirements in longitudinal direction. Figure 2.6 depicts the relationship between velocity v (in m/s) and the distance d (in m) to the pedestrian in front. Similar to Seyfried et al. [24], we find a linear relationship between those quantities:

$$d = 1.5v + 0.0035. \tag{2.3}$$

Summarizing, we find that the amplitude of lateral oscillation is high for people walking slowly whereas the frequency of that oscillation is smaller compared to walking with normal speed. Finally, the distance to the pedestrian in front tends to be smaller when walking slowly.

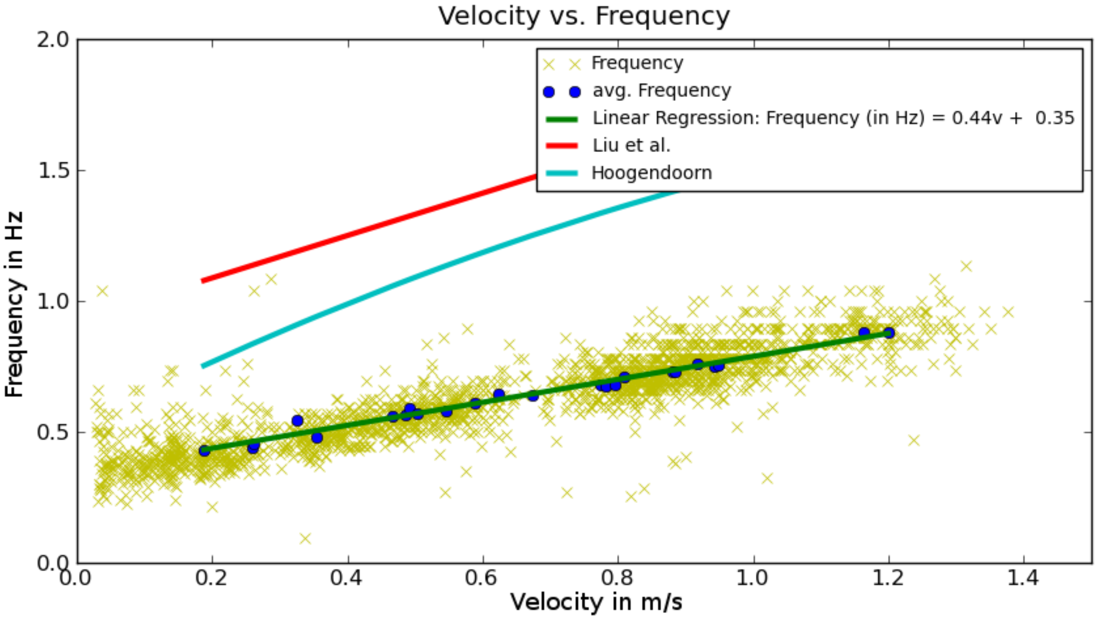


FIGURE 2.5: Relationship between velocity and frequency of lateral oscillation in comparison to the findings of Liu et al. [30] as well as Hoogendoorn and Daamen [37]. Blue markers correspond to mean velocity and mean frequency of different pedestrian groups.

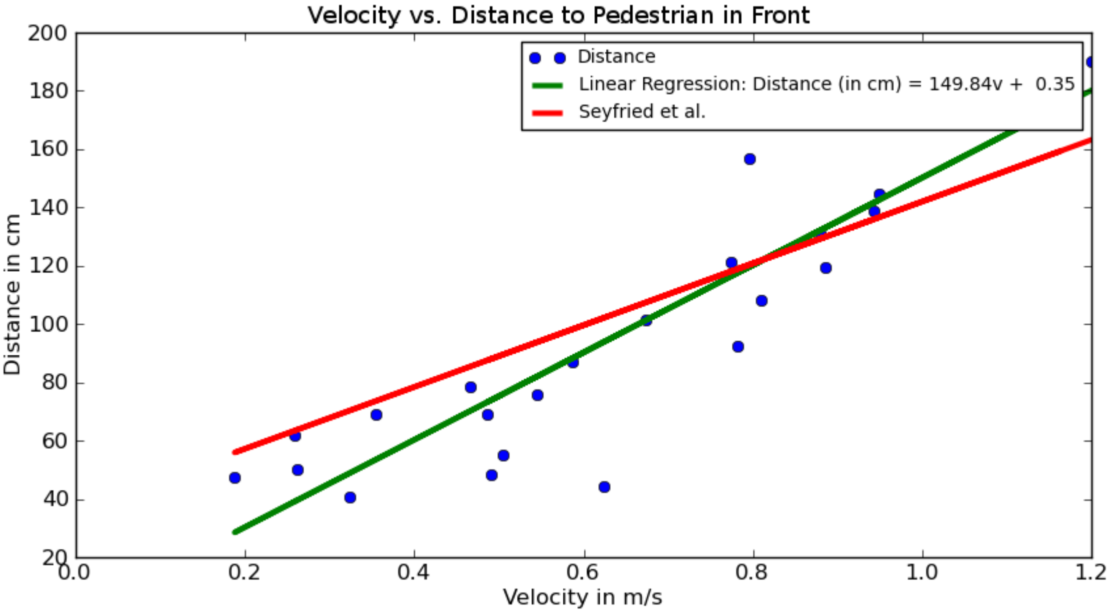


FIGURE 2.6: Relationship between mean velocity and the distance to the pedestrian in front in comparison to the findings of Seyfried et al. [24].

2.3 Pedestrian Behavior in Crowd Turbulences

Due to the lack of data, dynamics in crowd turbulences are not well understood. In fact, the work of Helbing and Johansson [9, 88] is the only empirical study of this crowd phenomenon. In video recordings of pilgrim flows in Mina, they observe a sudden transition from laminar flow to *stop-and-go waves* in areas without any obvious bottlenecks. These waves are characterized by intermittent flow of pedestrians and can last over a long period of time (over 20 minutes). According to Helbing et al. [9], stop-and-go waves are first signs of critically high pedestrian densities.

A second transition to even more critical motion patterns called *crowd turbulences* is observed when the flow becomes irregular. It is caused by people moving involuntarily inducing sudden movements of other people nearby. People loose control of their movements, are pushed around and fall down. They are trampled down and, moreover, they turn into obstacles for others leading to more stumbling people. As a result, most people die by suffocating due to dangerous pressure of up to 4500 N/m on their chests [2].

In contrast to vehicular traffic, the average velocity does not drop down to zero at high densities. Instead, people keep moving as studies of individual trajectories show. Moreover, at high densities pedestrian trajectories show backward motion and irregular motion into all directions.

Currently, traditional pedestrian simulation models do not handle mass behavior at extreme densities. The only work that takes the aforementioned observations into account is presented by Yu and Johansson [28]. They propose an extension of the social force model where fluctuations in the distances between pedestrians lead to strong repulsive forces causing sudden displacements. By simulating a situation that is comparable to the crowd disaster in Mina [9] and measuring the pressure, they show that their proposed model reproduces crowd turbulences adequately.

In order to assess the riskiness in a crowd, Helbing et al. [9] suggest to measure the pressure at a location as the product of local density times local velocity variance. Because of appropriate camera viewpoints, they are able to measure this quantity on video recordings of a crowd disaster using digital image processing techniques. They identify locations of high pressure that are indeed areas where people stumbled and suffocated. However, pedestrian detection and tracking in real-time is not possible in most applications due to inappropriate camera viewpoints, privacy concerns, as well as performance issues.

Summarizing, studies of crowd turbulences show that shock waves propagating through the crowd are characterized by sudden movements of people, irregular motion into all directions, as well as high variances in local velocities.

Chapter 3

Detection of Dangerous Mass Behavior

As previously pointed out, the pedestrian density in human crowds might reach a critical level leading to terrible stampedes. Table 1.1 lists some examples of recent crowd disasters and illustrates the need for research and new technologies in pedestrian dynamics and crowd management. Simulations of pedestrian flow and other means of crowd management such as video surveillance might help to avoid dangerous situations in crowds. However, security personnel at train stations, airports, mass events and other pedestrian facilities are faced with the short attention span of humans limiting their ability to monitor multiple video streams over a long time period. Wallace et al. [5] investigate the effectiveness of surveillance systems and compile guidelines for operators. They conclude that humans can effectively monitor screens for a time interval of thirty minutes to two hours. Moreover, most humans can only observe up to 4 camera screens. Here, it would be useful to have real-time surveillance systems analyzing multiple video streams automatically in the background and alerting security personnel if necessary.

Although researchers have conducted many experiments in order to understand the mindset of people in large crowds, the lack of data of real incidents limits research to use data obtained from experiments in which humans do not behave naturally. Up to now, the work of Helbing and Johansson [9, 31] is the only study of a real incident. Now, the video data of the Loveparade stampede that happened in 2010 and the reconstruction of the chain of events that led to disaster give valuable insights into human behavior at high densities.

In Section 3.1, we give an overview of this tragic incident. In order to develop automatic methods for detection of critical situations, we conduct a series of video analysis experiments on video footage of the Loveparade stampede in Section 3.2 and identify

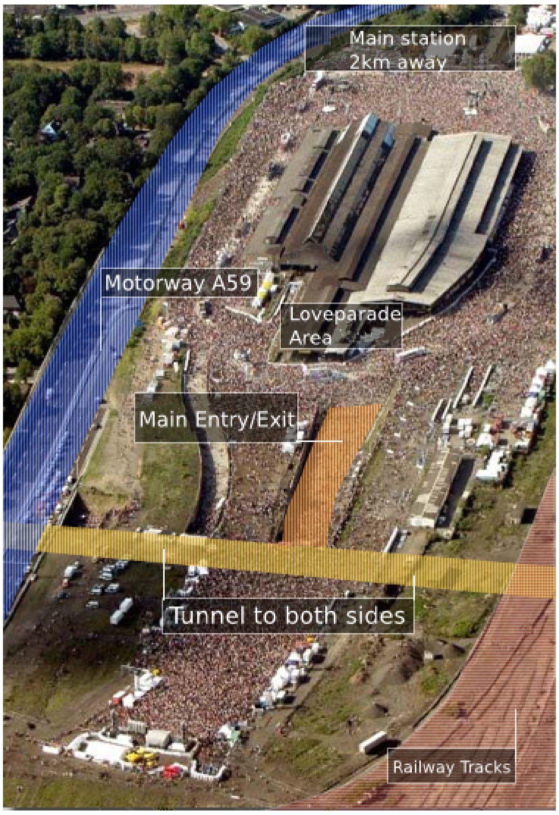


FIGURE 3.1: Aerial view of the festival area [89].

basic motion patterns that are characteristic for congestions. Based on our findings, we develop methods for the detection of congestions (Section 3.3) and crowd turbulences (Section 3.4).

3.1 Loveparade Stampede

On July 24, 2010, a German music festival, the Loveparade in Duisburg ended in a terrible stampede where 21 visitors died and more than 500 were injured. As of this writing, many questions as to the reasons of this catastrophe remain unanswered. However, it is unquestionable that at some locations of the festival area there were too many people present to guarantee their safety. At the main entry ramp to the festival area, the density of visitors was temporarily so extremely high that people who fell were trampled down by other visitors. All victims suffocated or were crushed to death due to enormous pressure on their chests.

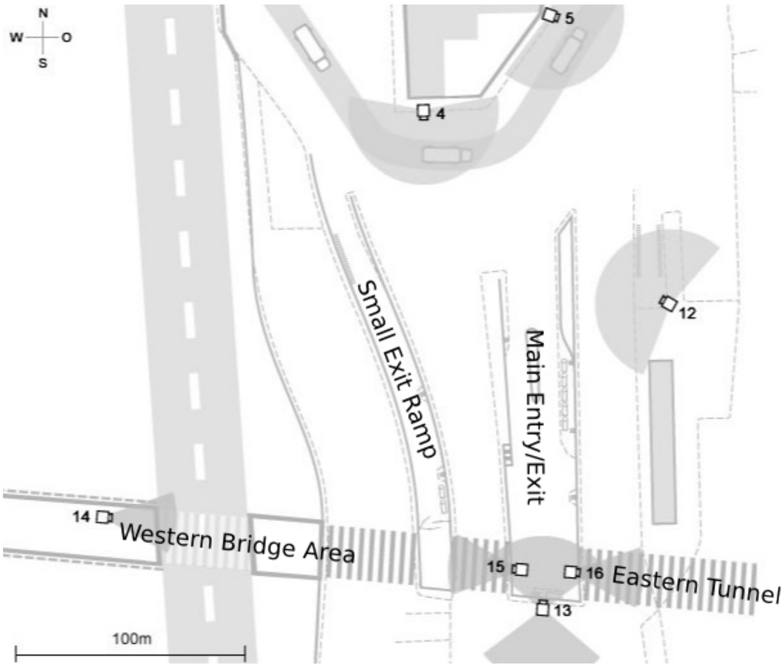


Figure 3.2: Schematic view of the festival area and locations of surveillance cameras [1].

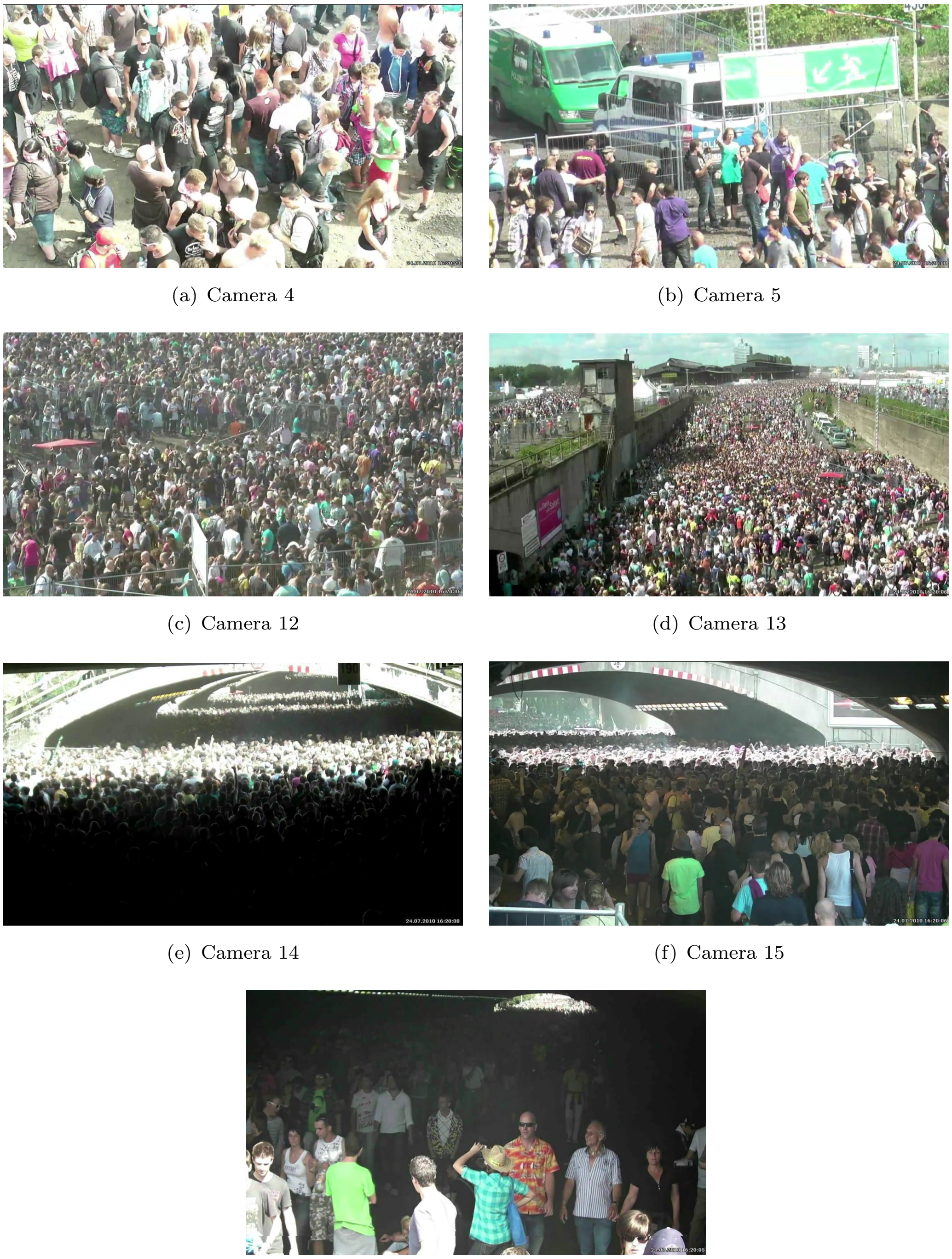
3.1.1 Festival Area

The Loveparade took place at the compound of the former freight station of Duisburg which is located near the city center. It is situated about 2km from the main station and lies between railway tracks and a highway. The former freight station has an area of about 230 000 square meters and about half of the total area was accessible to the visitors [90]. Figure 3.1 shows an aerial view of the festival area during the event and Figure 3.2 shows a schematic view of the compound. During the event, there were two entries for the visitors: The eastern tunnel which is about 250m long and the western bridge area. Both meet at the main ramp which simultaneously served as the main entry to- and the exit from the festival area. Another smaller exit ramp did exist, but had been closed by the police during the event.

3.1.2 Surveillance Cameras

The festival area was continuously monitored by seven cameras. Figure 3.2 shows the locations of each camera. Exemplary screenshots of all cameras can be seen in Figure 3.3. The entrance area in the eastern tunnel and the western bridge area were monitored by 4 cameras (referred to as camera 13 – 16) where three of them were static cameras (camera 14–16). Cameras 4, 5 and 12 were non-static cameras and monitored the upper part of the main festival area. Video footage recorded by all cameras can be downloaded from

[1]. In order to protect the privacy of the victims, video recordings are only available for the time between 13:30 and 16:40.



(a) Camera 4

(b) Camera 5

(c) Camera 12

(d) Camera 13

(e) Camera 14

(f) Camera 15

(g) Camera 16

Figure 3.3: Views of the scenes as recorded by the surveillance cameras [1]. See Figure 3.2 for the locations of the cameras. In the work reported here, we concentrate on cameras 14 and 15 for the detection of congestions and on camera 13 for the detection of shock waves.

3.1.3 Chronology

In the following, we summarize the chronology of the Loveparade [1]. For a timeline depicting the most important events, see Figure 3.4.

The festival area was opened for visitors at 12:00 and the festival began at 14:00. 30 minutes later, the number of visitors started to increase at the main entry/exit ramp. At 15:00, the security guards at the tunnel entrances closed some access control points thus reducing the number of visitors in the western bridge area. At 15:50, a first police cordon was formed in the western bridge area. At the same time, the small exit ramp was closed. A crowd in front of the small exit ramp resulted from visitors trying to leave the area. Between 15:50 and 15:57, a second police cordon was formed at the entrance of the eastern tunnel. At 16:00, the western access control points were closed, so that only a few visitors were visible in the western bridge area. Two minutes later, the access control point was re-opened resulting in a sudden increase of the number of visitors in the western bridge area. A crowd started to form behind the police cordon which had been established near the small exit ramp in the western bridge area. At the same time, a third police cordon was built on the main entry/exit ramp stopping those visitors that wanted to leave the festival area as well as arriving visitors that wanted to enter the main festival area. Only a few visitors were visible in the lower part of the main ramp, whereas in the western bridge area the crowd increased. Starting around 16:09, visitors being stopped by the third police cordon created a bottleneck on the main ramp. At 16:13, the small exit ramp was opened as an entry ramp. Due to the police cordon in the western bridge area, the large amount of visitors could hardly move. Some of the visitors began to climb a fence. At 16:15 the second police cordon in the eastern tunnel dissolved and the number of visitors in the lower part of the ramp increased. Five minutes later, the first police cordon in the western bridge area dissolved as well. The huge crowd that has been blocked by this cordon, suddenly moved to the lower part of the ramp where they met the visitors from the eastern tunnel. The visitor flows from both entries increased, but because of the third police cordon, they could not enter the festival area. Thus, the pressure on the third police cordon increased and it was dissolved at 16:24. Some visitors began to escape by climbing a small flight of stairs at the bottom of the ramp as well as light poles. A fourth police cordon was formed at the upper part of the main ramp increasing the pressure in the lower part. Starting from 16:40, the situation got out of control. More and more people tried to escape by climbing the stairs and light poles. In the following, 21 visitors died, more than 500 visitors were injured.

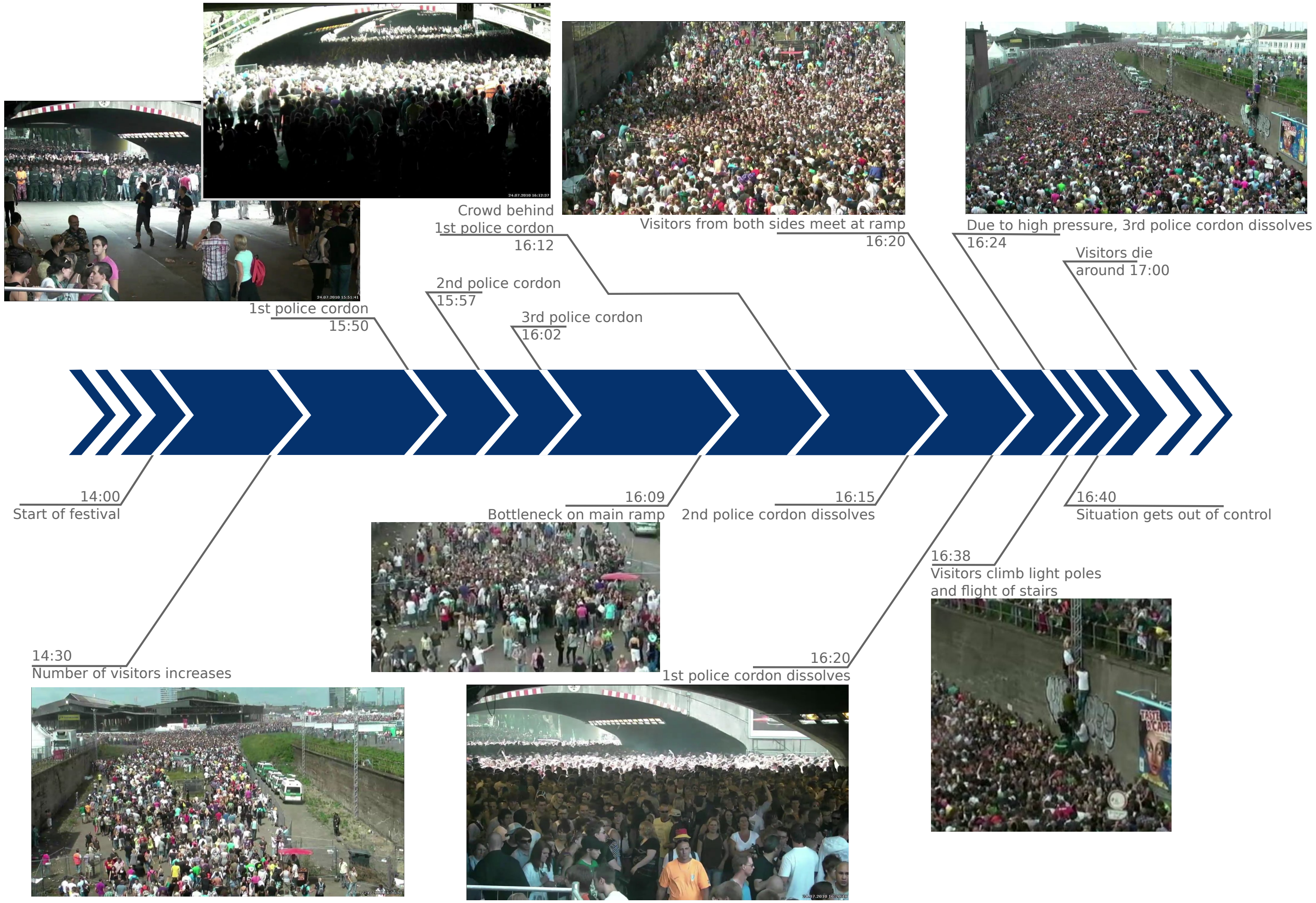


Figure 3.4: Timeline of the chronology [1].

3.2 Motion Patterns in Congested Areas

In order to develop automatic methods for detecting dangerous situations in crowds, we analyze video footage of the Loveparade stampede with the help of computer vision techniques. Here, we aim at detecting phase changes caused by changing pedestrian flow. In particular, we are interested in changing pedestrian flow due to congestions.

For our analysis, we concentrate on video footage from camera 15 which monitored the western bridge area. Figure 3.5 shows a screenshot which also depicts the main walking direction of the people walking to the festival area.

We first test optical flow as a feature for differentiating different motion patterns. However, in our scenario, motion patterns of a large crowd are composed of different actions performed by many different people. Therefore, we decompose complex motion patterns into simpler motion patterns (Section 3.2.2). Some of these motion patterns are very characteristic and, in Section 3.2.3, we relate these motion patterns to human walking characteristics.

3.2.1 Histograms of Dense Optical Flow Fields

In our first analysis, we investigate the utility of dense optical flow fields as a feature to represent global crowd movements. In particular, we investigate its utility to differentiate crowd motion of different time points in order to detect phase transitions. For optical flow computation, we apply the method proposed by Farnebäck [91] who uses quadratic polynomials to estimate translations of a local neighborhood and determines motion vectors from polynomial expansion coefficients. Figure 3.5 shows an example of the resulting optical flow fields for camera 15.

Given the dense optical flow field for a frame of a video sequence, we compute two-dimensional histograms of motion magnitude and motion angle of the flow vectors of the entire frame. All histograms are of size $36 \cdot 100$ where the magnitude ranges from 0 to 20 pixels divided into 100 bins and the motion orientation is divided into 36 bins. In order to test if optical flow is useful for distinguishing different time points and thus for detecting phase changes, we cluster all of the resulting two-dimensional histograms using the k-means algorithm. Figure 3.6 shows results for $k = 5$.

Figure 3.6(a) depicts how the individual frames of the video footage of camera 15 are assigned to different clusters. Here, different colors correspond to different cluster centers. Up until about 15:50, most frames (colored in blue in Figure 3.6(a)) correspond to the cluster whose centroid is shown in Figure 3.6(b). This cluster of flow vector histograms

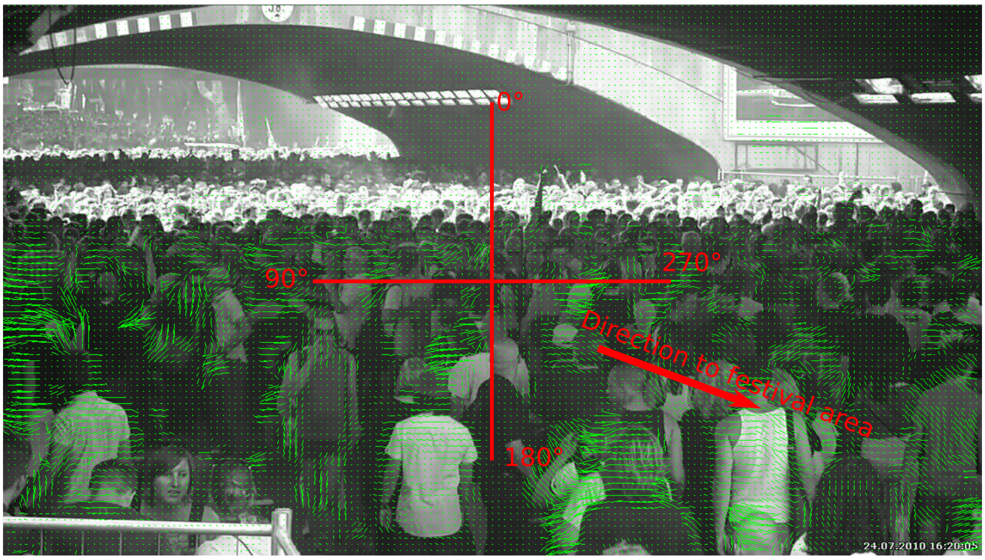
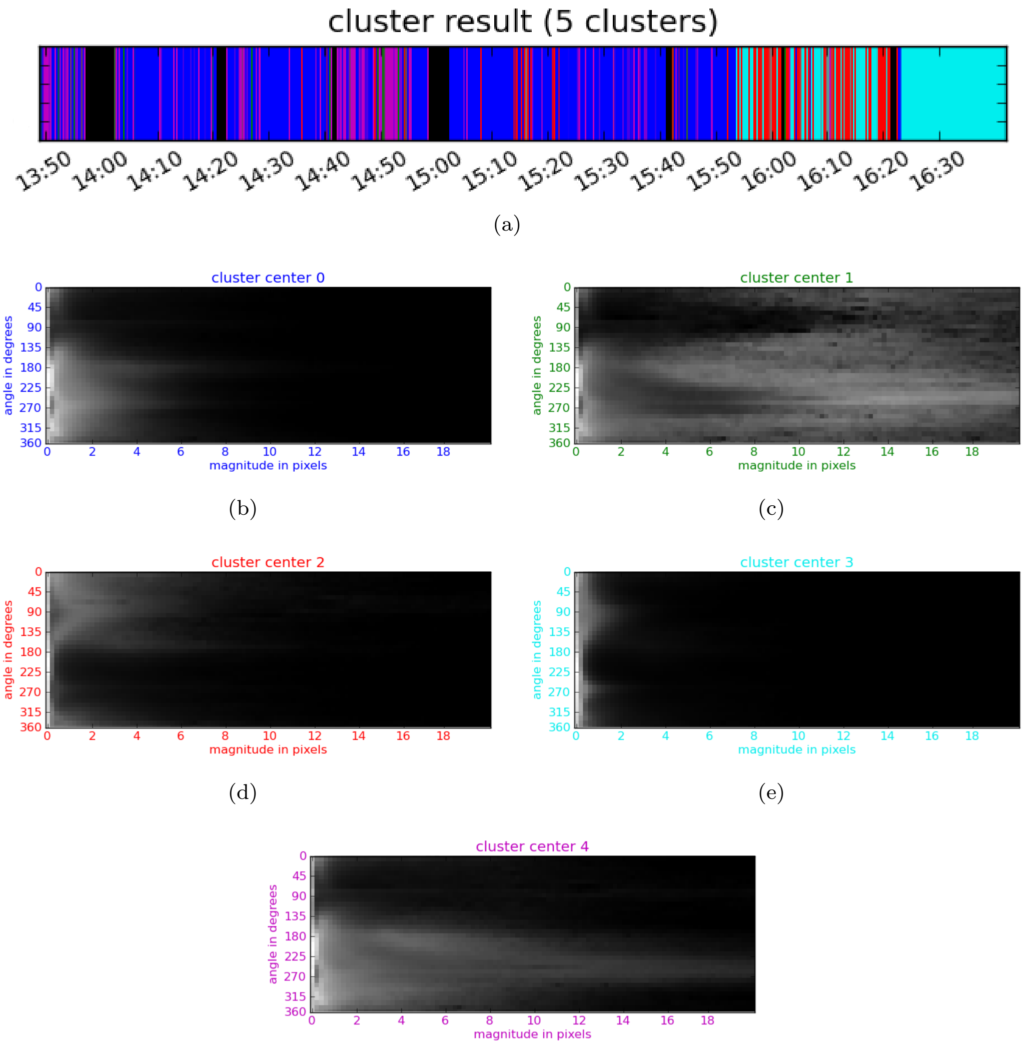


FIGURE 3.5: Dense optical flow computed using the method proposed by Farnebäck [91]. Green lines encode the optical flow where the motion angle is encoded by the angle of the line and the length of the line denotes the motion magnitude. The main walking directions of the people marching towards the festival area is also depicted in this figure.

represents a general motion to the right (around 250°) of a rather high magnitude: large numbers of people are marching towards the festival area. Shortly after 16:20, however, all histograms are assigned to the cluster (colored in cyan) whose centroid is shown in Figure 3.6(e). This cluster of motion vector histograms is characterized by weak motion oscillating between right (270°) and left (90°). This pattern is indicative of a congestion where there is no globally dominant motion direction anymore; instead of moving forward, the crowd has come to a halt and people are stepping from one foot to the other in order to keep their balance.

From the results of these initial feasibility analysis, we conclude that 2D histograms of optical flow vectors provide a suitable means to characterize basic motion patterns. As the analysis shows, this feature can distinguish crowd motion from different time points enabling us to discriminate phases of different crowd behaviors.



(a)

(b)

(c)

(d)

(e)

(f)

FIGURE 3.6: k-means clustering ($k = 5$) of two-dimensional histograms of motion angle and magnitude in an entire frame. Figure (a) shows the cluster assignment (encoded in colors) for camera 15 and Figures (b) to (f) the corresponding cluster centers. Most frames are assigned to the cluster center encoded in blue and depicted in Figure (b) which shows considerably more motion mostly in a direction of 250°. Shortly after 16:20, the frames are assigned to the cluster centroid depicted in Figure (e). This cluster contains vectors of little motion, mostly in directions 90°and 270°.

3.2.2 Extracting Characteristic Histograms

Having identified two-dimensional optical flow histograms as a suitable means to characterize crowd motion and to discriminate different crowd motion patterns, we now aim at understanding the crowd's behavior. Human motion can be considered as the superposition of atomic or basic actions and motion patterns. For example, an action like "walking" is composed of "swinging the left arm", "swinging the right arm", "stepping forward with the left leg", "stepping forward with the right leg". On a coarser level, the behavior of multiple humans on a platform in a subway station, for instance, is the superposition of people performing actions like “buying a ticket”, “sitting on a seat”, “standing at the platform”, or“going to the train”. Now, understanding human actions and behavior both on a finer and on a coarser level requires the recognition of atomic actions and their spatio-temporal relationship.

However, in our scenario, it is very difficult or even impossible to extract motion behavior of single humans visible in the scene. Thus, we consider complex motion that can be observed in a large crowd as the superposition of various motion behavior performed by different people and aim at extracting motion behavior of groups of people that perform similar actions. For that purpose, we examine complex motion patterns and decompose them into basic motion patterns.

Non-negative matrix factorization (NMF) is a suitable means to decompose matrices into meaningful parts. In [92], for instance, Lee and Seung apply NMF to images of faces and show that, in contrast to other decomposition methods, NMF basis images correspond to parts of faces. In action recognition, non-negative matrix factorization and other matrix decomposition methods have successfully been applied [57, 93] and shown to extract meaningful parts of human poses.

In our case, we use NMF to decompose complex motion patterns into a set of simpler motion patterns. Since motion varies over space and time, we divide the video sequence both spatially and temporally. We superimpose a grid of cells over the video frames where grid cells towards the back of the scene are smaller in order to account for perspective distortions, see Figure 3.7. For each cell of the grid, we again compute two-dimensional histograms ($36 \cdot 100$ bins) of motion angle and magnitude for the whole video sequence. Then, we apply non-negative matrix factorization to sets of histograms obtained during time intervals of 30 seconds. The set of m optical flow histograms is regarded as an $n \times m$ matrix V in which all entries are non-negative and where each column vector corresponds to a flattened histogram of size n, i.e. the columns of the two-dimensional histogram are concatenated to form a vector of size n. NMF finds an approximate factorization of $V \approx WH$ as

$$V_{i\mu} \approx (WH)_{i\mu} = \sum_{a=1}^{r} W_{ia} H_{a\mu}, 1 \leq i \leq n, 1 \leq \mu \leq m \tag{3.1}$$

In our case, the r columns of W represent flattened two-dimensional basis histograms of motion magnitude and motion angle. Each column of H consists of the coefficients for linearly combining the basis histograms in order to reconstruct the original histograms of V. W and H are of dimensions $n \times r$ and $r \times m$, respectively, where r is the number of basis histograms to be computed and is set to five in our case. Each of the r basis

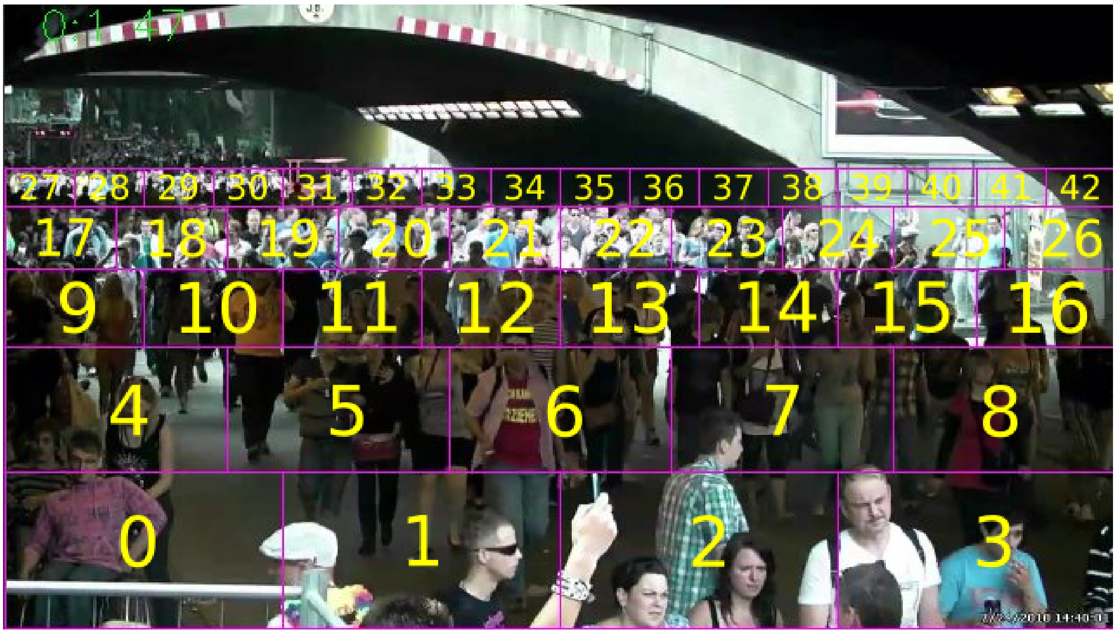


FIGURE 3.7: Superposition of a grid over the video frames where grid cells towards the back of the scene are smaller accounting for perspective distortions.

histograms describes motion patterns that are characteristic for certain parts of the scene as well as for particular time intervals. We use the multiplicative update rules of Lee and Seung [92] given in Equation 3.3 in order to minimize the objective function:

$$F = \sum_{i=1}^{n} \sum_{\mu=1}^{m} [V_{i\mu} \log(WH)_{i\mu} - (WH)_{i\mu}] \tag{3.2}$$

where W and H are non-negative. Because of this constraint, NMF learns parts-based representations which are additively combined to form a whole. The multiplicative update rules are given as [92]:

$$\begin{aligned} W_{ia} &\leftarrow W_{ia} \sum_{\mu} \frac{V_{i\mu}}{(WH)_{i\mu}} H_{a\mu} \\ W_{ia} &\leftarrow \frac{W_{ia}}{\sum_j W_{ja}} \\ H_{a\mu} &\leftarrow H_{a\mu} \sum_i W_{ia} \frac{V_{i\mu}}{(WH)_{i\mu}}. \end{aligned} \tag{3.3}$$

Figures 3.8 to 3.10 show basis histograms for three different time intervals: The basis histograms shown in Figure 3.8 represent typical motion patterns at 14:40. At that time, the flow of pedestrians is normal, visitors proceed to the main entry without encountering major obstacles. In the histograms, this is reflected by a sharp peak corresponding to a magnitude of 3 to 4 pixels and a motion angle of about 250° which corresponds to the direction of the main entry of the festival area, see Figure 3.5.

Figure 3.9 shows basic motion patterns at 15:53 shortly after the police cordon had been established. Now, most visitors are blocked and just a few continue moving towards the festival area. In the extracted basis histograms, a distinct dominant motion is not visible anymore.

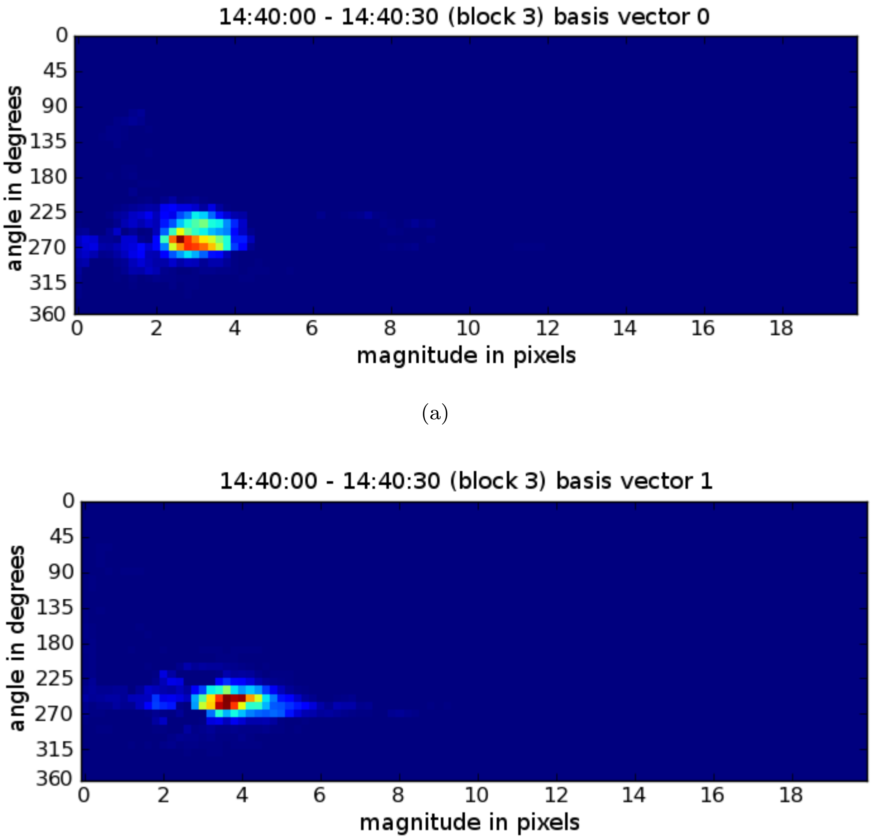
On the other hand, Figure 3.10 shows two motion patterns that are characteristic for the general crowd motion at about 16:28 when the visitor density is very high and causes congestion. The basis histogram in Figure 3.10(a) corresponds to rightward motion (270°); the histogram in Figure 3.10(b) reflects leftward motion (90°). Indeed, similar motion patterns showing both motion to the right and to the left can be observed for the whole period between 16:22 and 16:40 when the visitor density is over-critically high.

Given all the resulting basis histograms, we cluster them using the k-means algorithm ($k = 5$) and, for each time interval of 30 seconds, we determine the nearest cluster center. Figure 3.11(a) shows the cluster assignment for the entire video sequence. In time intervals of normal pedestrian flow, we observe motion patterns with a distinct peak at a magnitude of about 3 to 4 pixels and a motion angle of approximately 250° corresponding to the direction to the festival area (see Figures 3.11(b) and 3.11(f)). At 15:50, when the first police cordon is formed, no distinct motion is visible in the corresponding histograms in Figures 3.11(c) and 3.11(e). Contrarily, the motion patterns after 16:22 are composed of rightward and leftward motion as depicted in Figure 3.11(d).

3.2.3 Lateral Oscillating Motion in Congested Areas

The feasibility analysis described previously reveals several phases of characteristic motion patterns: In the first phase (between 13:40 and 15:50), we mainly observe motion angles of approx. 250° and a magnitude of 3-4 pixels. This pattern corresponds to movement of visitors into the direction of the main entry. The second phase begins at 15:50 when the first police cordon is formed in the western bridge area stopping the visitors. At that time, we cannot observe any motion into a distinct direction anymore. The last phase begins at about 16:22 shortly after the police cordon has been dissolved. The large crowd that was previously blocked from entering the festival area suddenly moves forward and quickly jams. In this congestion situation, the pedestrian density is very high and the only type of motion visible is to the right (270°) and to the left (90°) corresponding to a situation where people are stepping from one foot to the other in order to keep their balance resulting in oscillating motions.

In fact, such oscillating motions of a crowd are frequently observed in congestion situations and they result from the upper bodies of the people in the crowd swinging rightwards and leftwards. Similar to the studies of Hoogendoorn and Daamen [37] and



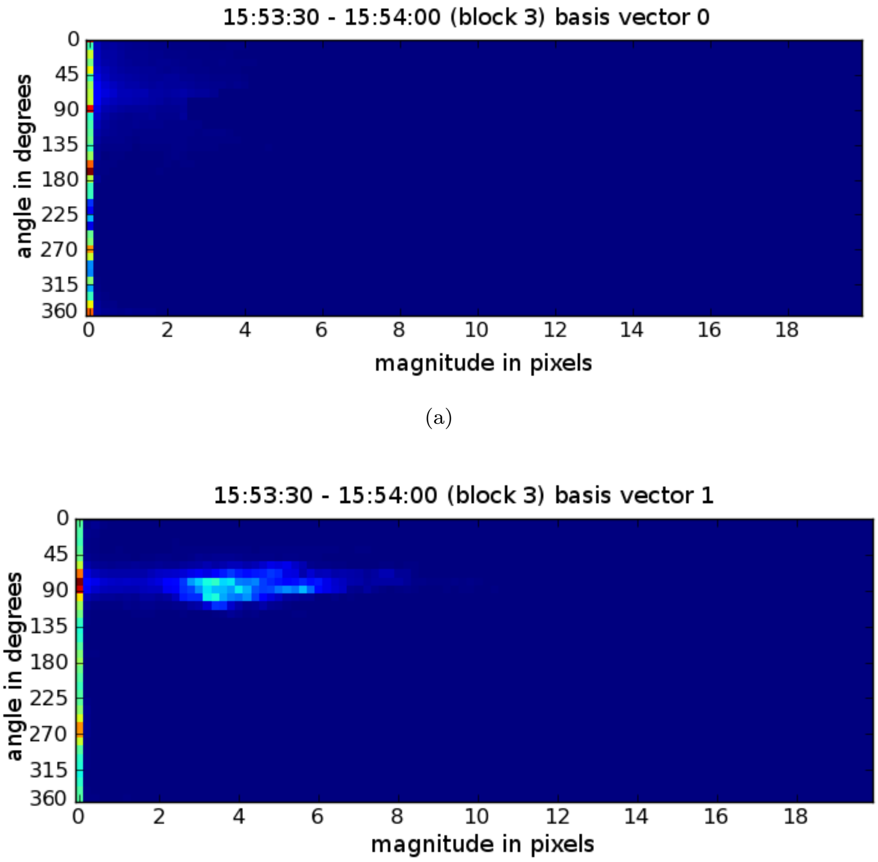
(a)

(b)

Figure 3.8: Two basis histograms extracted by applying NMF to 2D histograms of motion angle and magnitude obtained during a time interval of 30 seconds at 14:40 for a grid cell in the foreground. At that time, the pedestrian flow is normal. People are marching towards the festival area.

Liu et al. [30], we analyzed microscopic pedestrian behavior in congested areas on data obtained from a large scale experiment and observed lateral oscillation in the trajectories (see Section 2.2.2). This is caused by people who do not move along a straight line. Instead, it is a characteristic of human gait, that they tend to swing laterally. We observed that while the amplitude of the lateral oscillation is higher for lower speeds, the frequency increases for higher speeds. Equations 2.1 and 2.2 give linear relationships between the velocity and the amplitude as well as between the velocity and the frequency.

Summarizing, we conclude that, firstly, histograms of optical flow vectors are well suited for describing crowd motion. Changing pedestrian flow result in changing optical flow patterns enabling us to detect phase transitions. Secondly, from optical flow histograms we can infer the motion behavior of the crowd. More specifically, optical flow histograms during congestions which have characteristics described above originate from human walking patterns.



(a)

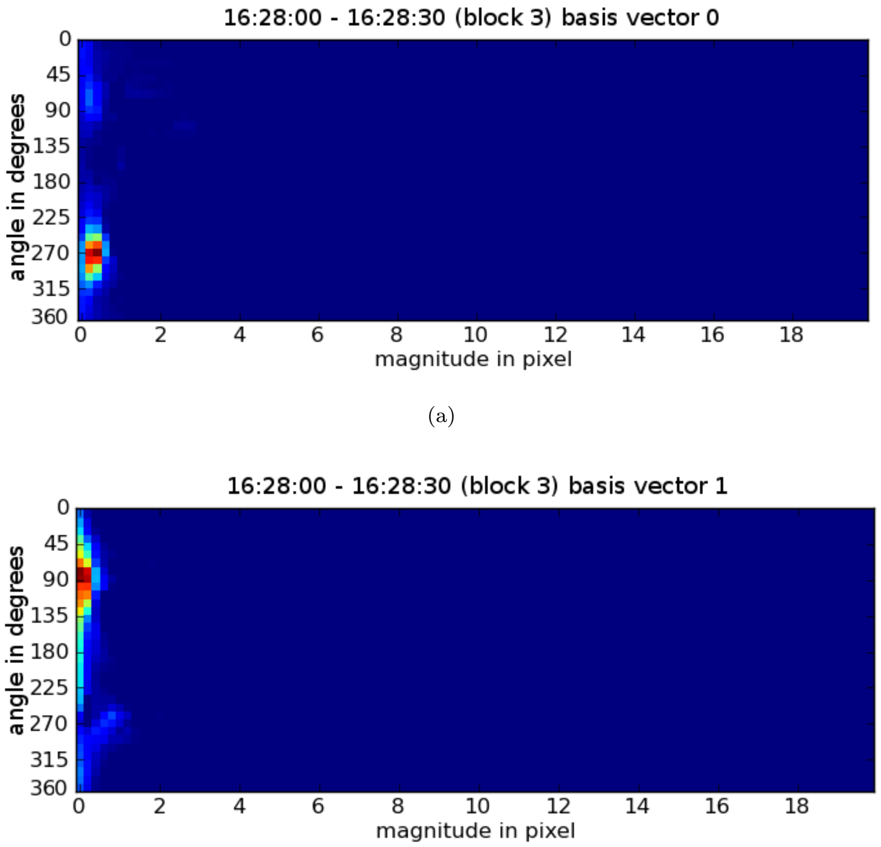
(b)

Figure 3.9: Two basis histograms extracted by applying NMF to 2D histograms of motion angle and magnitude obtained during a time interval of 30 seconds at 15:53 for a grid cell in the foreground. At that time, a police cordon stopped the visitors, so that only weak motion can be observed. Just a few visitors are not stopped by the police resulting in optical flow histograms such as Figure 3.9(b).

3.3 Detection of Congestions

Given the findings described in Section 3.2.3, we propose a method for the automatic detection of dangerous congestion situations. First, we compute dense optical flow and corresponding two-dimensional histograms of motion angle and magnitude. Then, we average the histogram over a short time interval t_{avg}. Histograms that are indicative of congestion situations show small motion along two major directions (rightwards and leftwards) which reflect lateral oscillation of the people's upper bodies. Such histograms show a high degree of symmetry (see Figure 3.11(d)) so that we measure the mirror symmetry of an optical flow histogram and consider the resulting value a feature for congestion detection.

Next, we apply a sequential change-point detection method to the extracted feature in order to detect phase changes which occur in situations of changing pedestrian flow.



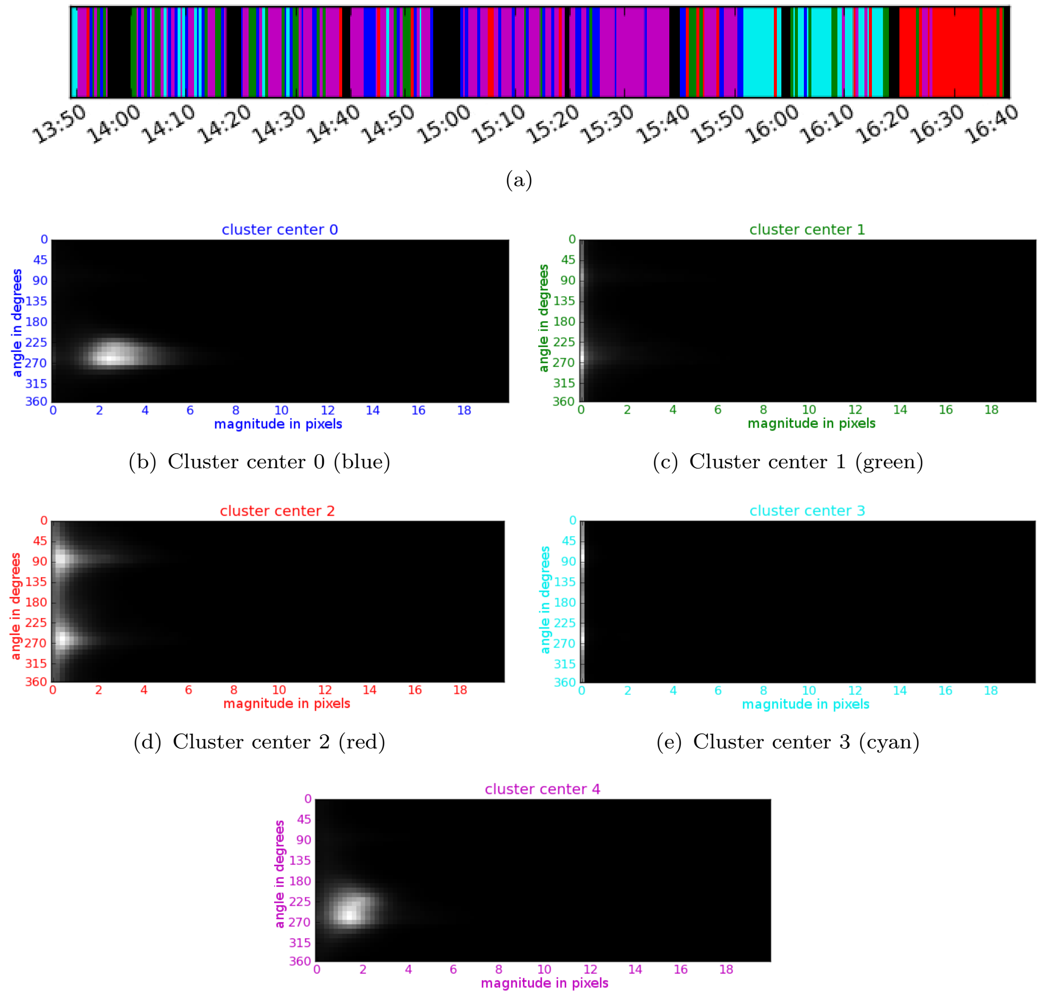
(a)

(b)

Figure 3.10: Two basis histograms extracted by applying NMF to 2D histograms of motion angle and magnitude obtained during a time interval of 30 seconds at 16:28 for a grid cell in the foreground. Figure (a) shows motion to the right (270°), whereas Figure (b) shows motion to the left (90°). At that time, the visitor density was very high.

Thus, we are able to detect unusual events which cause pedestrians to change their motion behavior. Since our proposed feature is designed to be sensitive to phase changes caused by increasing pedestrian densities, the system can detect congestions in crowds in particular.

In contrast to traditional approaches to real-time video surveillance, we do not attempt to detect and track individual people because the high densities of pedestrians and the challenging camera viewpoints that characterize crowded scenes would necessitate considerable computational effort. Instead, we consider dense optical flow fields to determine major motion patterns and motion directions in the crowd. In addition to computational efficiency, this also guarantees the privacy of people being monitored.



(a)

(b) Cluster center 0 (blue)

(c) Cluster center 1 (green)

(d) Cluster center 2 (red)

(e) Cluster center 3 (cyan)

(f) Cluster center 4 (purple)

FIGURE 3.11: k-means clustering ($k = 5$) of basis histograms obtained by applying NMF. Figure (a) depicts the cluster assignment for the entire video sequence. Figures (b) to (f) show the five cluster centers. The cluster centers in Figures (b) and (f) show distinct motion peaks with a high magnitude and a motion angle of approx. 250°. This motion pattern is characteristic for time intervals of normal pedestrian flow. Figures (c) and (e) do not exhibit any distinct motion; these basic motion patterns correspond to the situation observed after the police cordon has been established. The cluster center in Figure 3.11(d) shows left- and rightward motion. It is characteristic for motion observed between 16:22 and 16:40 when the pedestrian density is very high and people are stepping from one foot to the other in order to keep their balance.

3.3.1 Mirror Symmetry of Optical Flow Histograms

For detecting changes in pedestrian flow and congestions in particular, we compute a symmetry measure of a two-dimensional optical flow histogram by summing up the absolute differences between the histogram and a flipped version of itself as described in the following. We subdivide each frame into a set of cells with cells in the background of the scene being smaller to account for effects of viewing perspective. Let $H_{t,c}(dir, mag)$ be the two-dimensional histogram of angle and magnitude of cell c at time t. Then, flipping $H_{t,c}(dir, mag)$ around a horizontal symmetry axis and denoting it by $\hat{H}_{t,c}(dir, mag)$, we compute

$$sym_{t,c} = \sum_{dir,mag} w(dir) \cdot \left|\hat{H}_{t,c}(dir, mag) - H_{t,c}(dir, mag)\right|. \tag{3.4}$$

The horizontal axis for flipping $H_{t,c}(dir, mag)$ is set to 180°, see Figure 3.5 for angle orientations. However, in order to account for changes in viewpoint, we also compute $sym_{t,c}$ with $H_{t,c}(dir, mag)$ being flipped at other horizontal axes around 180° (e.g. 170° and 190°) and take the minimum of these values. Additionally, for emphasizing leftward and rightward motion, we introduce a weight $w(dir)$ computed as:

$$w(dir) = \begin{cases} \dfrac{|dir - 90°|}{10°} & dir \leq 180° \\ \dfrac{|dir - 270°|}{10°} & dir > 180° \end{cases} \tag{3.5}$$

The weight for 90° and 270° is 0 whereas it has its maximum value of 9 for 0°, 180° and 360°. Accordingly, low values of $sym_{t,c}$ indicate that $H_{t,c}(dir, mag)$ is highly mirror-symmetric with peaks around 90° and 270° and is indicative for an obstructed pedestrian flow in a congested area.

3.3.2 Sequential Change-Point Detection

While low values $sym_{t,c}$ indicate mirror-symmetric optical flow histograms with peaks around 90° and 270°, the scale of the this measure is not clear. It highly depends on different factors, such as the camera viewpoint, the number of people visible in the scene as well as the environment.

In order to overcome this problem, we learn typical values of $sym_{t,c}$ and detect deviations for identifying anomalies. Thereto, we apply a sequential change-point detection algorithm enabling us to detect unusual events as well as congestions in particular. The method proposed by De Oca et al. [94] extends the conventional cusum algorithm [95]. It is a non-parametric cusum algorithm that allows for distributions varying in time and uses historical data for obtaining suitable thresholds above which an alarm is raised. We extend this algorithm to compute an additional measure that characterizes the severity of an alarm.

Let Y denote the value of $sym_{t,c}$. Now, consider a sequence of observations $\{Y_i\}_{i=1}^{N}$. We use previous observations $\{Y_j\}_{j=i-k-l}^{i-k}$ to estimate a reference distribution where k is a fixed time interval and l is a fixed number of historical observations that are used for estimating the reference distribution. Next, we denote the upper and lower α-percentiles of the reference distribution as $Q(\alpha)$ and $Q(1-\alpha)$, respectively, where α is specified by the user and controls the degree a deviation from the reference distribution is considered as critical. See Figure 3.12 for an illustration.

The cusum algorithm continuously accumulates deviations of incoming observations from the reference distribution:

$$
\begin{aligned}
S_i^+ &= \max\{0, S_{i-1}^+ + Y_i - Q(\alpha)\}, S_0^+ = 0 \\
S_i^- &= \max\{0, S_{i-1}^- + Q(1-\alpha) - Y_i\}, S_0^- = 0
\end{aligned}
\tag{3.6}
$$

It raises an alarm if either $S_i^+ > \Theta$ or $S_i^- > \Theta$, where in the first case, we detect an upward shift of the signal and a downward shift in the latter case. The threshold Θ is calculated from the reference distribution as follows: Suppose that the sequence of observations is drawn from the reference distribution, that is, no anomaly occurs. Using a bootstrap resampling method, Θ is selected so that the probability of a false alarm is equal to γ, a parameter specified by the user. For that purpose, M sequences are sampled from the reference distribution. For each sampled sequence m, cusum statistics according to Equation 3.6 are computed and $\max\{S_{sampled,m}^+, S_{sampled,m}^-\}$ is determined. Next, for each sampled sequence m, we select the maximum value of $\max\{S_{sampled,m}^+, S_{sampled,m}^-\}$ and compute the threshold Θ as the $(1-\gamma)$-percentile from these maximum values.

Whenever either S_i^+ or S_i^- exceeds the computed threshold Θ, we raise an alarm (see Figure 3.13). De Oca et al. [94] also propose a method for detecting the end time of an alarm. They apply a slope testing technique for detecting a downward trend in the cusum statistics which indicates that the deviations from the reference distribution become smaller: Without loss of generality, we assume that S_i^+ exceeds Θ at time a (The same rationale holds for S_i^-.). Then, a linear regression model is continuously fitted to a sliding window of cusum values $\{S_i^+\}_{i=n-\nu+1}^{n}$ for $n = a, a+1, \ldots$ and ν being a fixed size of the sliding window, see Figure 3.14. The end time of an alarm is detected, when the slope of the linear regression model is less than or equal to zero. Then, cusum statistics S_i^+ or S_i^-, respectively, are set to zero. Additionally, we propose to measure the severity of the raised alarm as a value $L \in [0 \ldots 1]$ by computing the angle of the regression

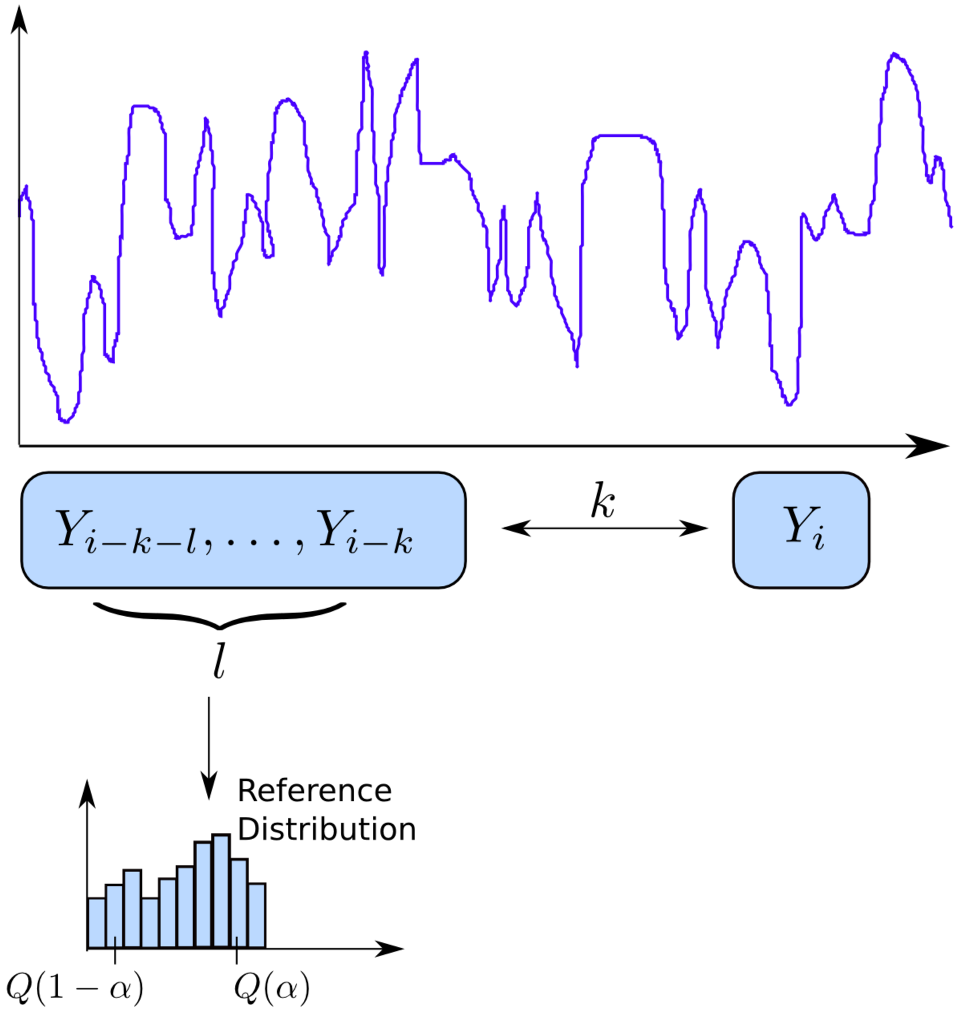


FIGURE 3.12: Given the current observation Y_i, a reference distribution is obtained from historical data $Y_{i-k-l}, \ldots, Y_{i-k}$. The upper and lower α-percentiles of the reference distribution ($Q(\alpha)$ and $Q(1-\alpha)$) are used in Equation 3.6 for calculating cusum statistics. Here, α controls the degree a deviation from $Q(\alpha)$ or $Q(1-\alpha)$ is considered as critical.

line in degrees and dividing it by 90°. This is motivated by the idea that the slope of the linear regression model of the cusum statistics S_i^+ (or S_i^-, respectively) depends on the deviation of the current observation Y_i to the reference distribution: The higher the deviation is compared to the reference distribution, the larger the slope of the regression line is. If L is near to one, the slope is large and the situation is considered to be very critical.

We apply the sequential change-point detection algorithm to the mirror symmetry measure proposed in Section 3.3.1 in order to detect phase transitions and congestions in

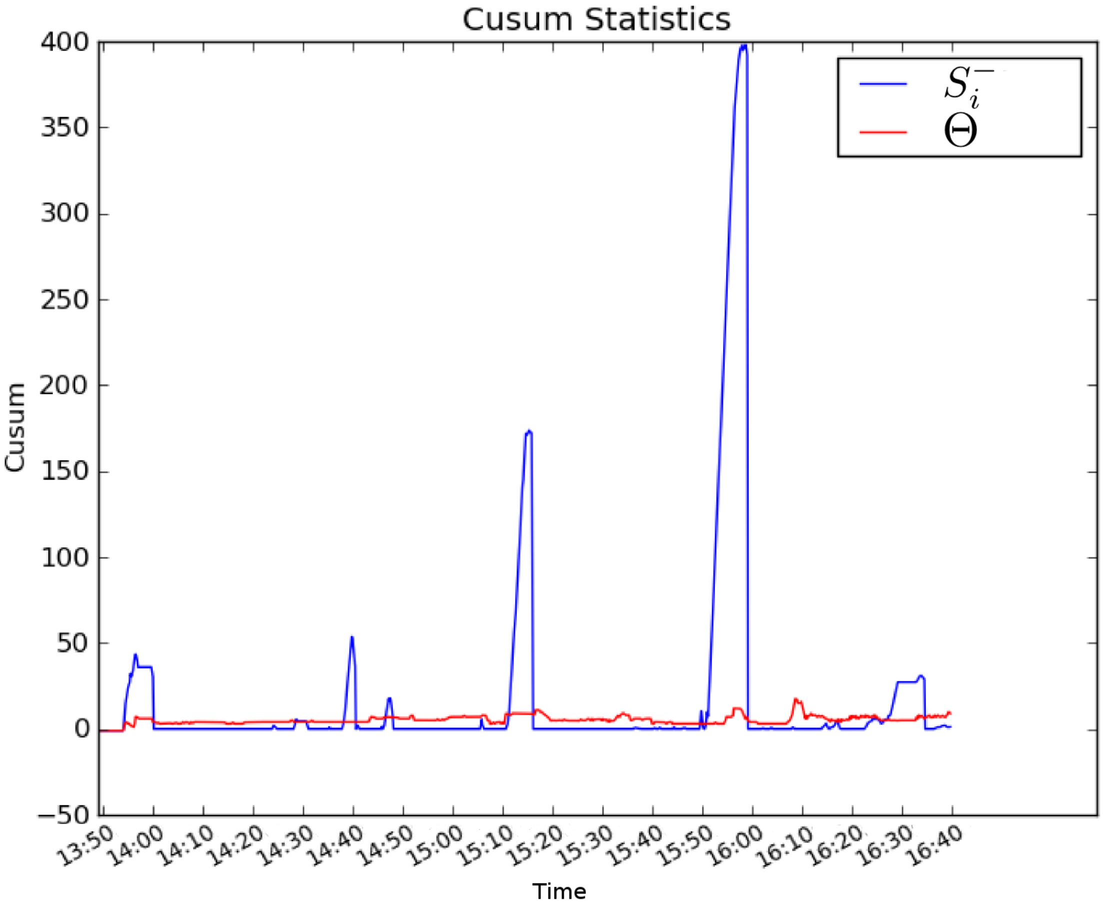


FIGURE 3.13: Cusum statistics S_i^- calculated on the $sym_{t,c}$ feature values obtained from the Loveparade dataset (Section 4.1.4, camera 15). Whenever S_i^- exceeds the threshold Θ, an alarm is raised. The threshold Θ has been determined automatically by using a bootstrap resampling method.

particular. Congestions are characterized by low values of $sym_{t,c}$ as described above. Thus, an alarm raised by the system is very severe and may indicate a congestion, if L is near to one and S_i^- exceeds the threshold indicating that $sym_{t,c}$ decreases due to optical flow histograms becoming more and more mirror-symmetric. Sections 4.1.3 and 4.1.4 give results obtained from video sequences showing congested areas.

3.4 Detection of Crowd Turbulences

In areas of extremely high pedestrian density, the movement of a person affects other nearby people and crowd turbulences might occur. As described in Section 2.3, in crowd turbulences, the pedestrian density is critically high and the variance in velocities increases. Therefore, Helbing et al. [9] measure the pressure exerted on the pedestrians as the product of local density times local velocity variance.

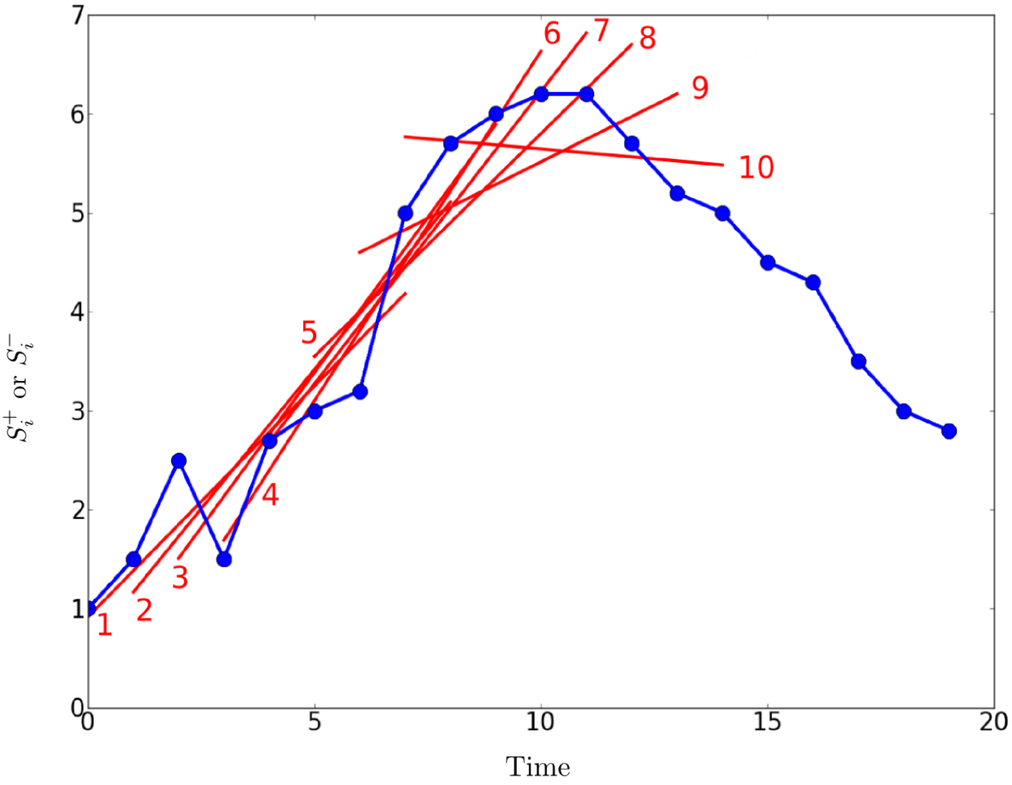


FIGURE 3.14: Slope test for cusum statistics. At each time a, a linear regression model is fitted to a sliding window of cusum values $\{S_i^+\}_{i=n-\nu+1}^{n}$ for $n = a, a+1, \ldots$ (here: $\nu = 8$). The end time of the alarm is detected at time $t_a = 12$, when the slope of the regression line (labeled as regression line 10) is less than zero for the first time. Additionally, we measure the severity of an alarm as the angle of the regression line divided by 90°.

Our analysis of video footage from the Loveparade stampede confirms these observations. The videos show short sequences of shock waves in the vicinity of the main entry which was monitored by camera 13 (a non-static camera). In these video sequences, we observe a sudden increase of the magnitude of the optical flow motion vectors as well as a high standard deviation of the magnitude of flow vectors as some people in a local region are already pushed by the wave whereas other nearby people do not move. Moreover, since several people in a local neighborhood are pushed into the same direction, the standard deviation of local motion angles is small.

Given these observations, we propose a method for the automatic detection of crowd turbulences. Thereto, we compute dense optical flow fields as described in Section 3.2.1. Then, we divide the frame into a grid of C cells. For each cell c and each time t, we compute the average magnitude of optical flow motion vectors $\mu_{mag,t,c}$, the standard deviation of the magnitude of optical flow motion vectors $\sigma_{mag,t,c}$ and the standard deviation of motion angles $\sigma_{dir,t,c}$. As described above, shock waves are characterized by a high magnitude and a high standard deviation of motion magnitudes as well as a small standard deviation of motion angles. Hence, we also compute a value $p_{t,c} =$

$\frac{\mu_{mag,t,c} \cdot \sigma_{mag,t,c}}{\sigma_{dir,t,c}}$ for each cell c and each time t. In those cells of a frame where a shock wave is observed to propagate, $p_{t,c}$ will be high. In order to normalize the feature $p_{t,c}$, we compute $\mu_{row,t}$, the average value of all $p_{t,c}$ values in a row of the grid as well as the standard deviation $\sigma_{row,t}$ of all $p_{t,c}$ values in a row. Then, we normalize each $p_{t,c}$ by subtracting the mean value of the corresponding row and dividing by the standard deviation all $p_{t,c}$ values in the row:

$$p_{t,c}^{norm} = \frac{p_{t,c} - \mu_{row,t}}{\sigma_{row,t}} \tag{3.7}$$

Now, we apply the sequential change-point detection algorithm described in Section 3.3.2 to learn typical values of $p_{t,c}^{norm}$ for each cell c and detect deviations from that value in order to detect shock waves. See Section 4.2 for results on the Loveparade dataset.

Chapter 4

Experiments

4.1 Congestion Detection

The feature for congestion detection presented in Section 3.3.1 has a low value in congested areas where the pedestrian flow is obstructed. However, the scale of this quantity is not clear and depends on different factors (e.g. camera viewpoint, environment, number of people). Hence, we learn typical values of the feature value and detect deviations in order to identify anomalies and congestions by applying a sequential change-point detection algorithm (see Section 3.3.2). Here, it is of vital importance that phases of different pedestrian flow result in different values of the feature. In the following, we verify this assumption by applying this feature to a synthetic dataset, a dataset obtained under laboratory conditions as well as two real world datasets.

4.1.1 Synthetic Dataset

We conduct a series of experiments on synthetic datasets and test different aspects of our feature. In particular, we investigate the sensitivity of our feature with respect to the distance of pedestrians to the camera, the velocity, the number of pedestrians and the camera viewpoint. In all experiments, we use the linear relationships (Equations 2.1 to 2.3) that we found by analyzing the trajectories obtained from a large scale experiment (Section 2.2.2) and generate artificial videos of pedestrians walking on ideal trajectories towards the camera.

In our first experiment, we generate synthetic videos of a single pedestrian walking with different speeds ($v = 0.2, 0.4, \ldots, 1.2, 1.4\frac{m}{s}$) on an ideal trajectory as described above towards the camera. We subdivide the frame into a grid of cells (see Figure 4.1) and compute the feature described in Section 3.3.1 based on optical flow histograms averaged

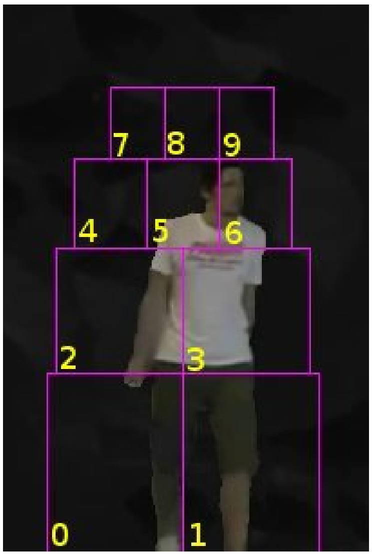


FIGURE 4.1: Superposition of a grid over the video frames where yellow numbers denote the number of the particular grid cell.

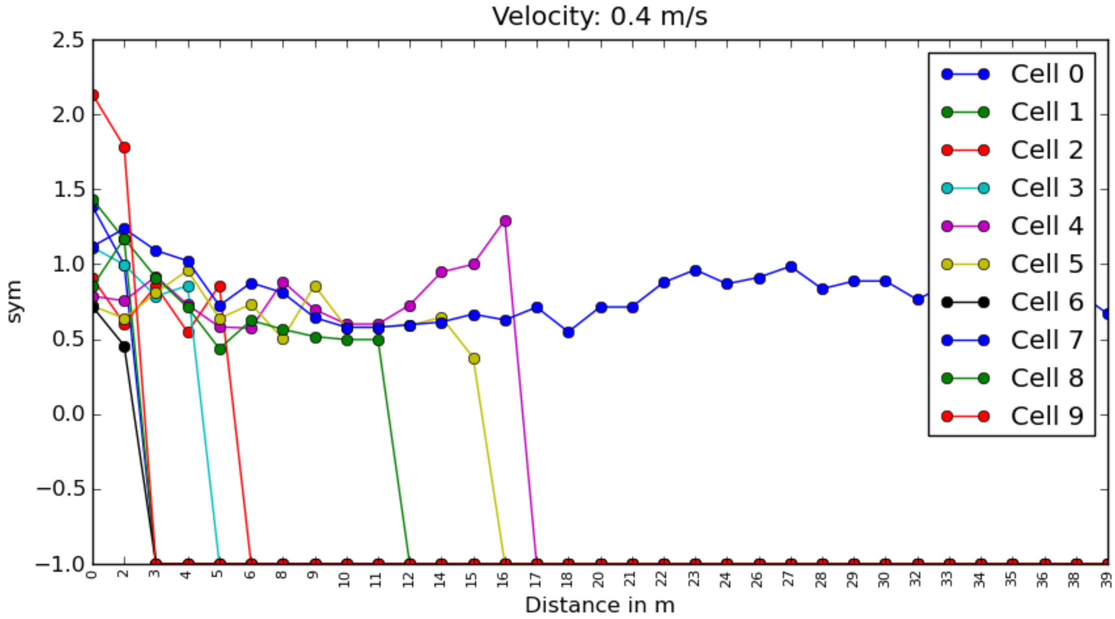


FIGURE 4.2: Relationship between feature values and distance to the camera for an artificial video sequence with a single pedestrian walking with a velocity of 0.4 m/s on an ideal trajectory towards the camera. One can observe that $sym_{t,c}$ increases to a value > 0 for different cells c as the pedestrian crosses cell c. For positions of the cells, see Figure 4.1. Note that we set $sym_{t,c}$ to -1 here, if no motion is observed in cell c at time i.

over different time intervals. Figure 4.2 shows the development of $sym_{t,c}$ depending on the distance of the pedestrian to the camera. The value of $sym_{t,c}$ increases for a cell c, when the pedestrian crosses the cell. Interestingly, the time offset of the curves for different cells is clearly visible.

Primarily, we are interested in the difference between the feature values for different

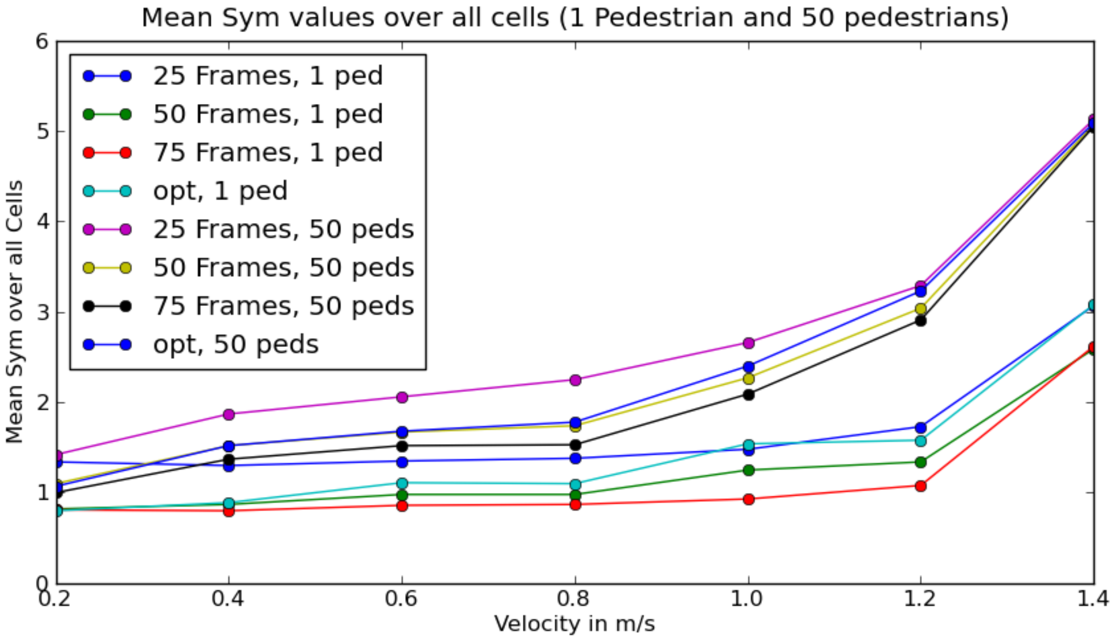


FIGURE 4.3: Mean feature values over all cells for a single pedestrian as well as for a group of 50 pedestrians walking with different velocities.

velocities. Figure 4.3 shows mean feature values over all cells for a single pedestrian as well as for a group of 50 pedestrians walking with different velocities. Obviously, the feature value is lower for lower velocities. In addition to that, we vary the time interval t_{avg} for averaging optical flow histograms (25, 50 and 75 frames). For the time interval denoted *opt*, we computed an optimal time interval for each velocity such that we can observe a full period of lateral oscillation. Note that this time interval depends on the velocity and ranges from 58 frames for 0.2 m/s to 26 frames for 1.4 m/s.

From Figure 4.3, we can observe that the difference in feature values between normal velocities (1.34 m/s) and velocities $\leq$ 0.6 m/s observable in congestions is increased when more pedestrians are visible. This is caused by the fact that more left and right movements are performed when multiple pedestrians are walking. Furthermore, Figure 4.3 reveals no significant differences between different time intervals for averaging optical flow histograms.

In the next experiment, we focus on phase transitions. We generated synthetic videos (25 fps) of 50 people walking fast and then abruptly decrease their speed. Again, we compute $sym_{t,c}$ for all cells c and sum up all values giving sym_t. Then, we compute the mean and the standard deviation of sym_t for the phase with high velocity, denoted as μ_1 and σ_1, as well as the mean of sym_t for the second phase of low velocities, denoted as μ_2. Now, we check if the values in the first phase differ significantly from values in the second phase by checking if $|\mu_1 - \mu_2| > 2 \cdot \sigma_1$. Table 4.1 gives results for different configurations. It shows that phases of different velocities are mapped to different ranges of feature

Phases $v_1 \Rightarrow v_2$	25 Frames	50 Frames	75 Frames
1.4 m/s ⇒ 0.6 m/s	✓	✓	✓
1.4 m/s ⇒ 0.4 m/s	✓	✓	✓
1.4 m/s ⇒ 0.2 m/s	✓	✓	✓
1.2 m/s ⇒ 0.6 m/s	✓	✓	✓
1.2 m/s ⇒ 0.4 m/s	✓	✓	✓
1.2 m/s ⇒ 0.2 m/s	✓	✓	✓
1.0 m/s ⇒ 0.6 m/s	✓	✓	✓
1.0 m/s ⇒ 0.4 m/s	✓	✓	✓
1.0 m/s ⇒ 0.2 m/s	✓	✓	✓
0.8 m/s ⇒ 0.6 m/s	-	-	-
0.8 m/s ⇒ 0.4 m/s	-	-	-
0.8 m/s ⇒ 0.2 m/s	✓	✓	✓

Table 4.1: Results obtained from artificial videos (25 fps) with two phases. In the first phase, pedestrians walk with velocity v_1 and decrease their velocity to v_2 in the second phase. We check if the values in the first phase differ significantly from the values of the second phase, i.e. if $|\mu_1 - \mu_2| > 2 \cdot \sigma_1$ where μ_1 and μ_2 are the mean feature values of the first and the second phase and σ_1 the standard deviation in the first phase. Moreover, we vary the time interval t_{avg} for averaging optical flow histograms from 25 frames to 75 frames.

values for most of the tested configurations. Importantly, the feature can distinguish between velocities observed in normal walking conditions (average walking speed 1.34 m/s [29]) and velocities typically observed in congestions ($\leq$ 0.6 m/s). Here, we do not observe any significant difference between different time intervals for averaging optical flow histograms.

In order to test the sensitivity of the feature with respect to the viewpoint, we generate synthetic videos with different camera viewpoints. Figure 4.4 shows a topview of the setup with four different camera positions and Figure 4.5 shows exemplary views from the four viewpoints.

Similar to the previous experiment, 50 pedestrians walk fast towards the camera, then abruptly decrease their speed. Again, we investigate the differences between the two phases by computing the mean feature values for both phases, μ_1 and μ_2, as well as the standard deviation of the first phase σ_1. Figure 4.6 depicts the ratio $\frac{|\mu_1 - \mu_2|}{\sigma_1}$. The higher this ratio is, the larger is the difference between the two phases facilitating the detection of the change point.

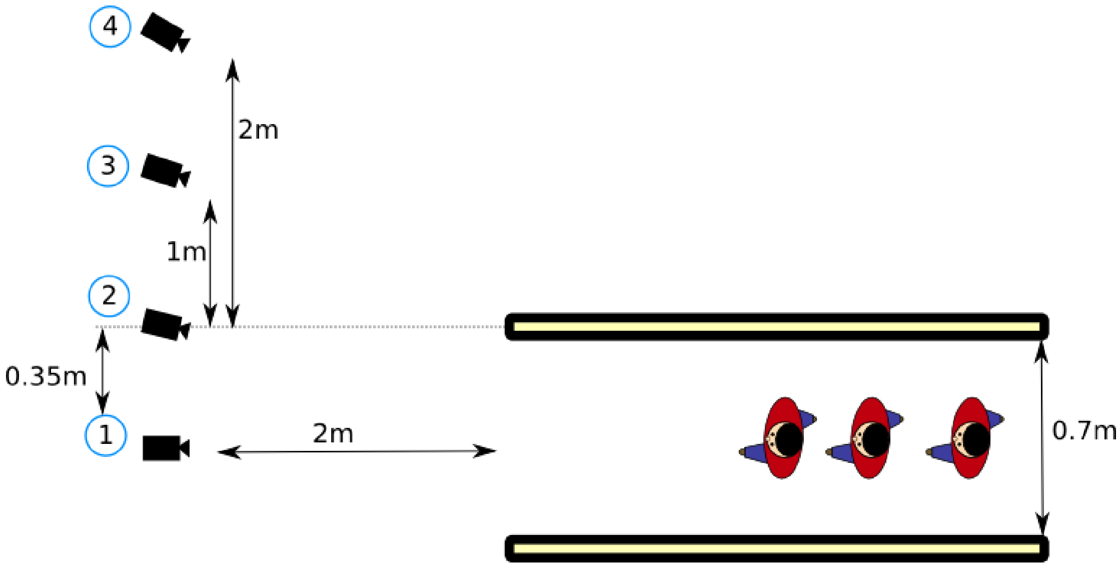


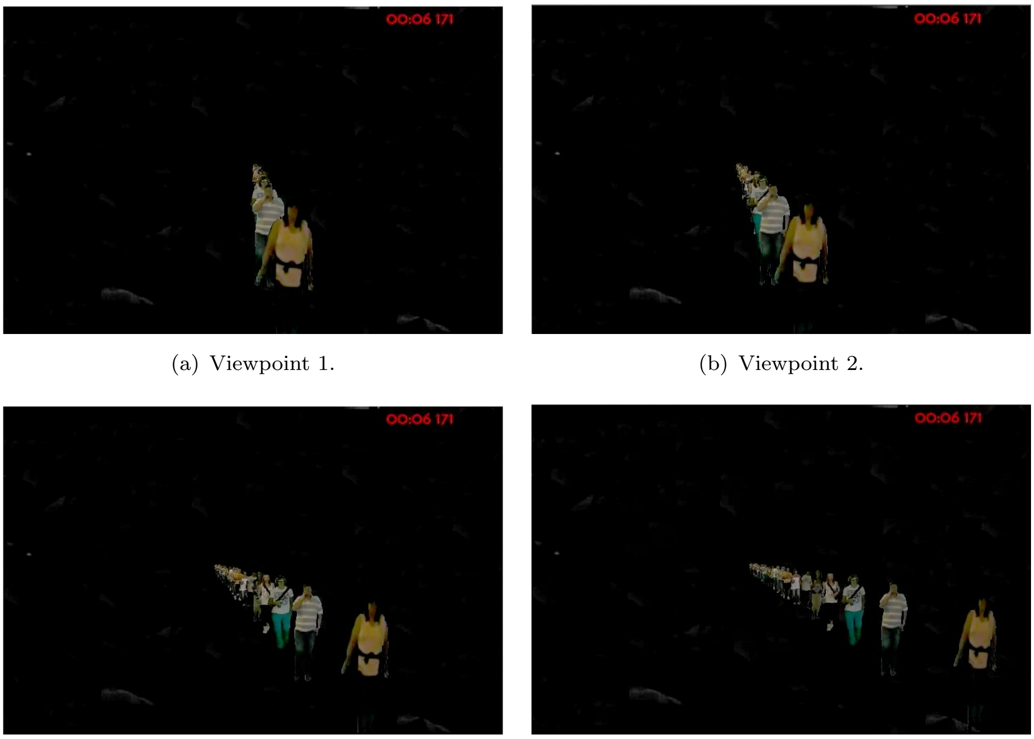
Figure 4.4: Topview of the experimental setup with camera positions for testing the relationship between our feature and camera viewpoint. In all experiments, the height of the camera is 2 m. See Figure 4.5 for exemplary views from the four different viewpoints

From Figure 4.6, one can observe that basically the ratio is higher when the speed of the pedestrians decrease to a very low level (0.2 m/s). Second, the ratio is always higher when averaging optical flow histograms over a longer time period. Next, comparing different viewpoints, we notice that the ratios are higher for viewpoints from the side (viewpoints 2, 3 and 4) compared to the central viewpoint 1. This may be due to the fact that from viewpoints 2, 3 and 4, more pedestrians are visible resulting in more left-to-right movements that are captured by the feature.

Summarizing, we conclude that feature values for different speeds differ significantly enabling us to detect change points automatically. The time interval t_{avg} for averaging optical flow histograms should be larger than 25 frames (for videos with 25 fps) in order to capture a full period of lateral oscillation. Finally, we notice that changes of pedestrian speeds can be observed from various viewpoints. Actually, observing a scene from a non-central viewpoint facilitates discrimination of different velocities, since more pedestrians are visible resulting in more left-to-right movements that can be observed.

4.1.2 Hermes Dataset

In our next experiment, we use video data of a large group of pedestrians walking through a corridor in order to test the relationship between the proposed feature and measures typically used in pedestrian dynamics. Second, we test the sensitivity of the feature regarding camera viewpoints and use the sequential change-point detection algorithm for detecting phase changes while studying the influence of different parameters. Finally, we



(a) Viewpoint 1.

(b) Viewpoint 2.

(c) Viewpoint 3.

(d) Viewpoint 4.

Figure 4.5: Exemplary views of synthetic videos from different viewpoints. See Figure 4.4 for the setup.

conduct experiments on video sequences recorded by topview cameras. For this purpose, we adapt the feature described in Section 3.3.1 in order to detect congestions from a topview.

The dataset has been recorded by the Hermes project [96] under laboratory conditions. Here, pedestrians walk through a corridor with a bottleneck at the end of the corridor resulting in different phases: First, pedestrians enter the corridor while walking fast (approx. at time 00:20 min). Then, due to the bottleneck, pedestrians have to lower their speed (approx. at time 00:40 min). Finally, since most pedestrians have passed the bottleneck, the speed increases again at time 02:00 min.

The corridor has a width of 3 m and is 8 m long and the bottleneck at the end has a width of 1.20 m. The scene is recorded by two overhead cameras with a small overlapping field of view and a frontal camera. For exemplary views, see Figure 4.7. By means of digital image processing [32], the trajectories of the pedestrians have been extracted within the Hermes project enabling us to determine measures such as velocity and pedestrian density.

For the frontal view, we superimpose a grid of cells over the video frame (see Figure 4.8(c)) and compute $sym_{t,c}$ values for each cell separately where we average optical flow

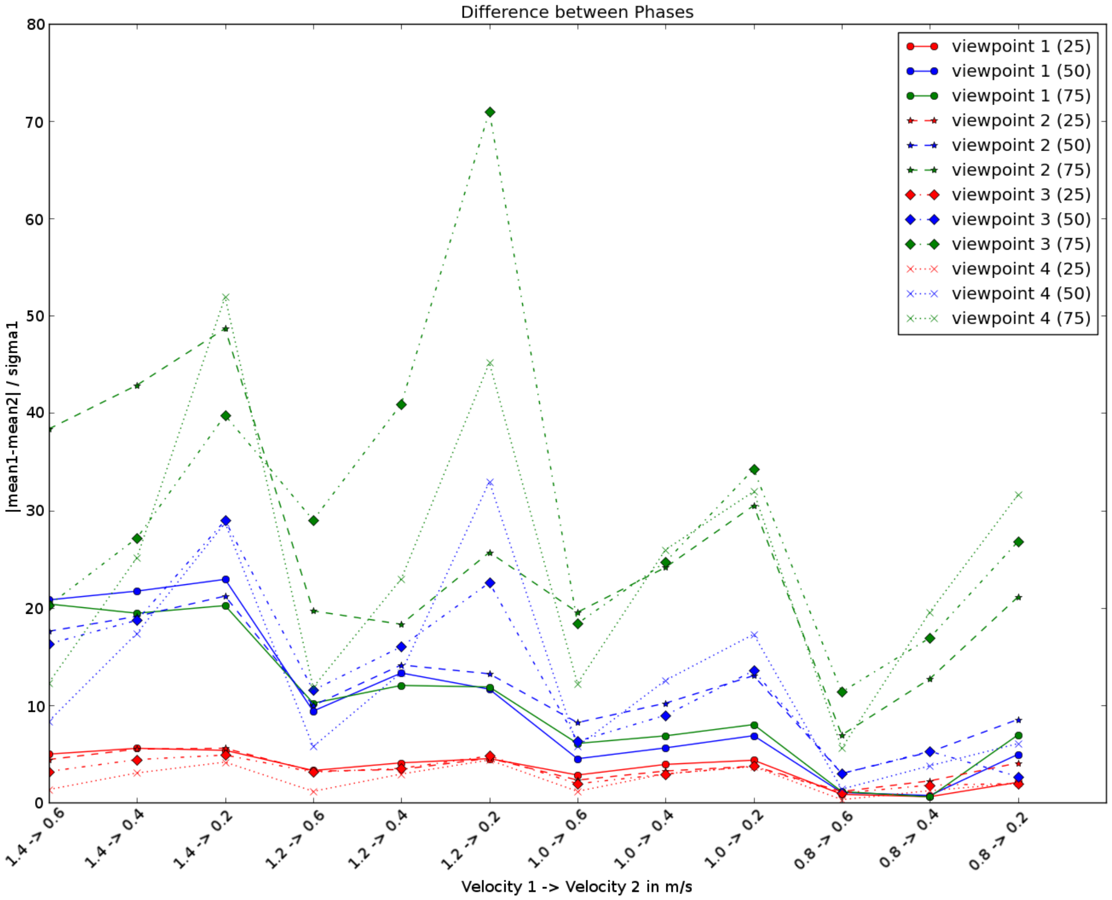


FIGURE 4.6: Ratio $\frac{|\mu_1 - \mu_2|}{\sigma_1}$ for different viewpoints, different time intervals for averaging optical flow histograms and different synthetic videos, where μ_1 and μ_2 are the mean feature values for the first and the second phase, respectively. σ_1 is the standard deviation of feature values in the first phase. Higher ratios indicate that the feature values in the first phase differ significantly from feature values in the second phase facilitating the automatic change point detection. For camera positions, see Figure 4.4

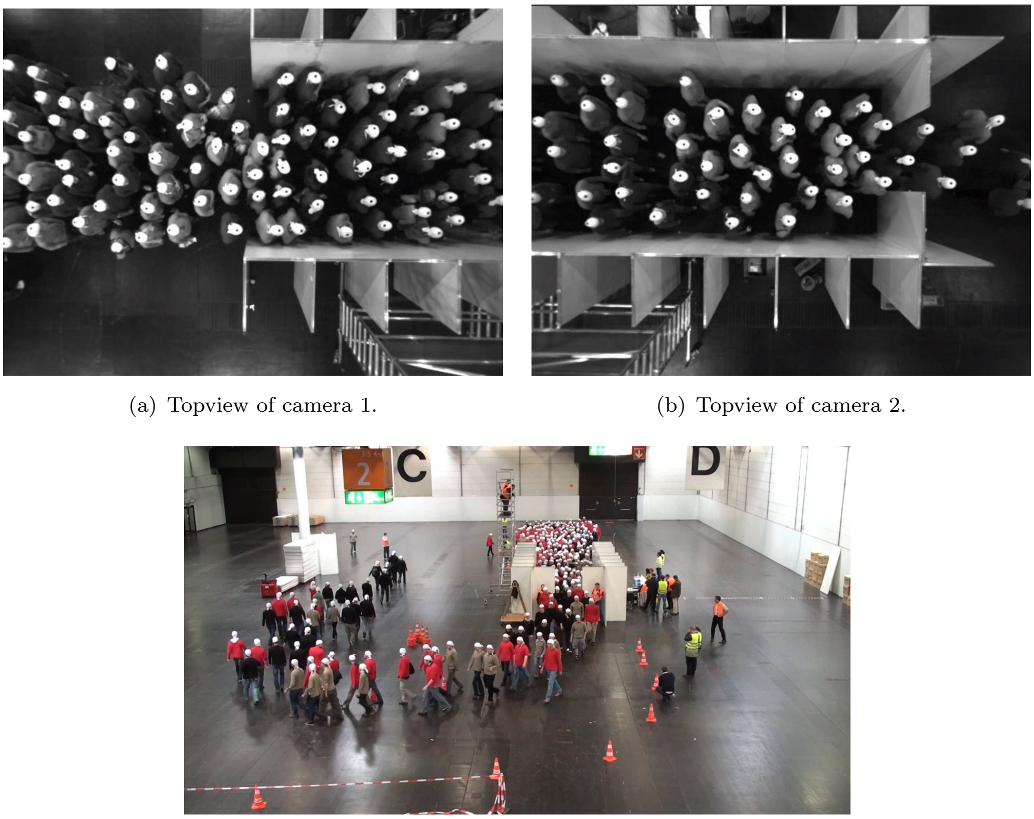
histograms over a time interval t_{avg} of 25, 50 or 75 frames, respectively (The video has a frame rate of 25 fps.). Then, we sum up $sym_{t,c}$ values of all cells and obtain sym_t. The development of sym_t is shown in Figure 4.8(b).

Next, we determine the average speed and pedestrian density over time. Given the trajectory $\vec{x}_i(t)$ of pedestrian i and a fixed Δt, the velocity of pedestrian i at time point t is computed as [84]:

$$\vec{v}_{\Delta t,i}(t) = \frac{\vec{x}_i(t + \Delta t/2) - \vec{x}_i(t - \Delta t/2)}{\Delta t}. \tag{4.1}$$

The average speed of N pedestrians is now

$$\bar{s}(t) = \frac{1}{N} \sum_i ||\vec{v}_{\Delta t,i}(t)|| \tag{4.2}$$



(a) Topview of camera 1.

(b) Topview of camera 2.

(c) Frontal view of camera 3.

FIGURE 4.7: Views of the experimental setup of the Hermes video dataset [96].

Figure 4.8(a) shows the average speed for the Hermes dataset over time where we set $\Delta t = 0.5s$.

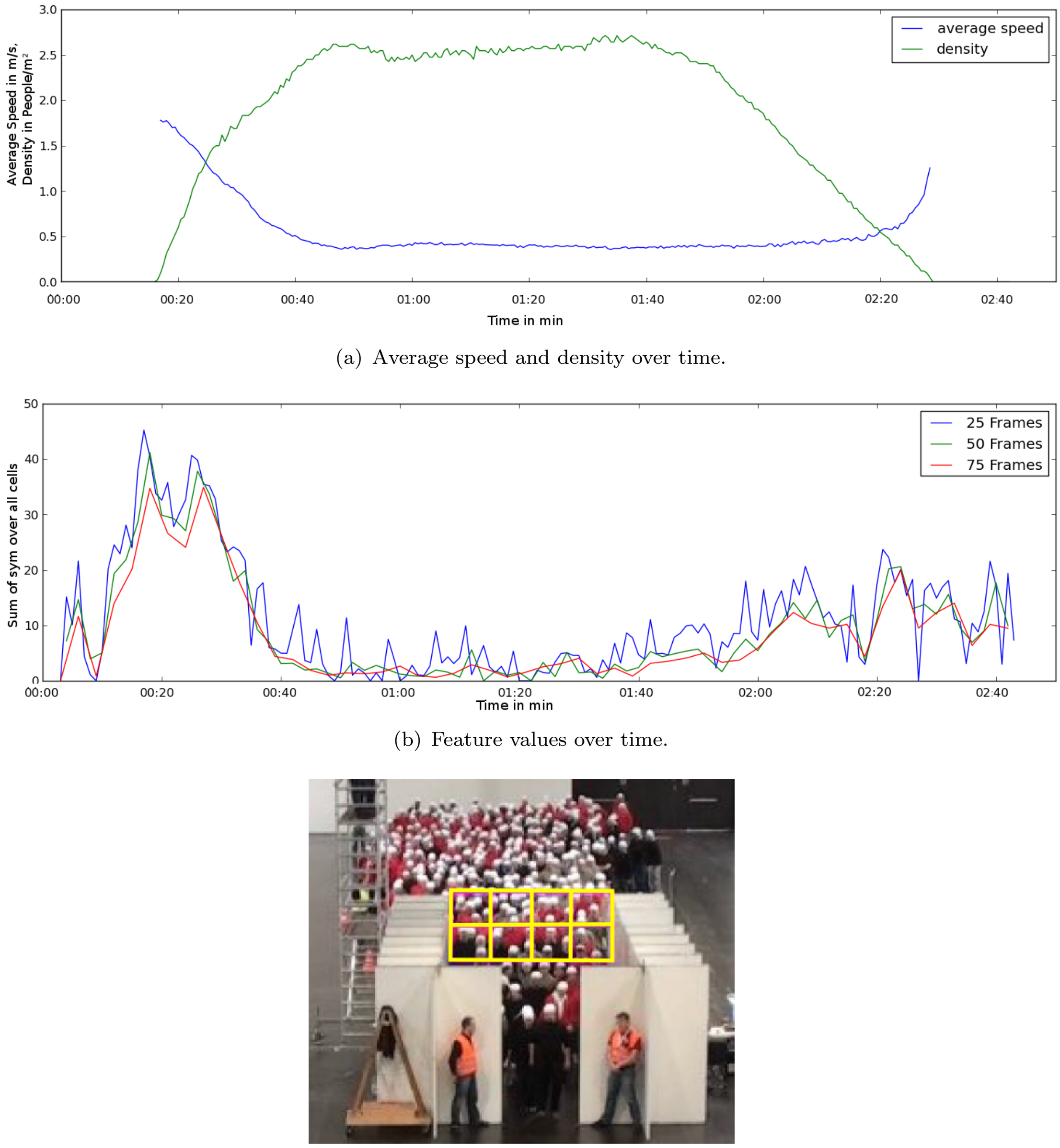
Pedestrian density is defined as number of pedestrians N per area $|A|$:

$$D = \frac{N}{|A|} \tag{4.3}$$

In Figure 4.8(a), we show the development of the density over time for the entire corridor.

Comparing sym_t values over time in Figure 4.8(b) and the curves in Figure 4.8(a) reveals substantial similarities. As the pedestrian density increases resulting in decreasing speeds, the feature value decreases indicating that we can observe left-to-right movements originating from lateral oscillations. In particular, after approx. 50 seconds, the density is at its highest level, whereas the feature value is at its lowest level. Then, at time 01:40 min, the density begins to decrease and, simultaneously, the feature values increase.

Figure 4.8(b) also shows that we obtain smoother curves when we average optical flow histograms over longer time periods.



(a) Average speed and density over time.

(b) Feature values over time.

(c) Superposition of grid.

FIGURE 4.8: Average speed and density of pedestrians over time in comparison with feature values for camera 3. We superimpose a grid of cells over the video frame, see Figure (c). Then, we compute feature values for each cell and sum up the values. Figure (b) shows the development over time for different time intervals. One can identify phases of different pedestrian flow. Figure (a) shows average speed and density of pedestrians over time. One can observe that the curve in Figure (b) decreases with decreasing speed and increasing density.

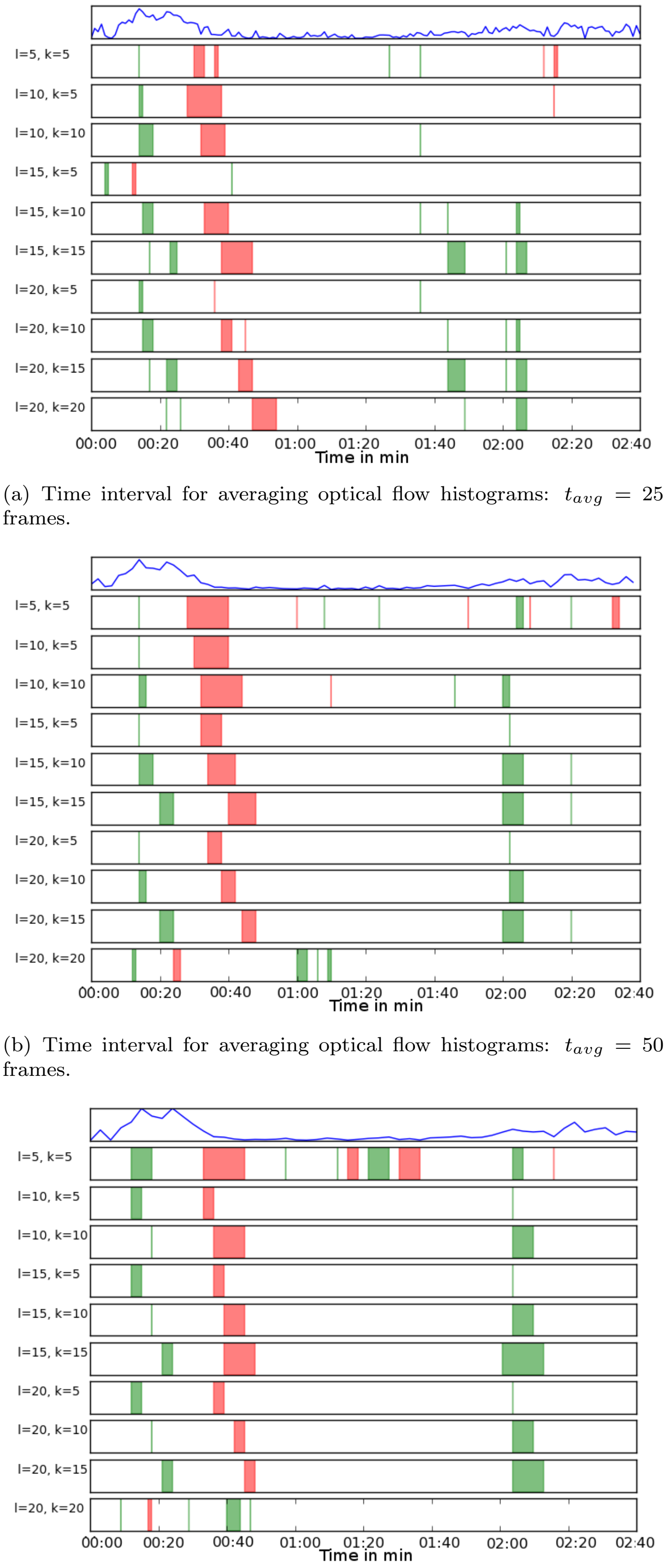
Parameters	$t_{avg} = 25$ Frames	$t_{avg} = 50$ Frames	$t_{avg} = 75$ Frames
l=5s, k=5s	0.43, 0.75, 0.54	0.3, 1, 0.46	0.33, 1, 0.5
l=10s, k=5s	0.66, 0.66, 0.66	1, 0.66, 0.8	1, 1, 1
l=10s, k=10s	0.66, 0.66, 0.66	0.6, 1, 0.75	1, 1, 1
l=15s, k=5s	0, 0, 0	1, 1, 1	1, 1, 1
l=15s, k=10s	0.6, 1, 0.75	1, 1, 1	1, 1, 1
l=15s, k=15s	0.83, 1, 0.9	1, 1, 1	1, 1, 1
l=20s, k=5s	0.66, 0.66, 0.66	1, 1, 1	1, 1, 1
l=20s, k=10s	0.83, 1, 0.9	1, 1, 1	1, 1, 1
l=20s, k=15s	0.83, 1, 0.9	1, 1, 1	1, 1, 1
l=20s, k=20s	0.8, 1, 0.8	0.2, 0.3, 0.25	0.2, 0.3, 0.25

Table 4.2: Precision, recall and F1 score for different parameters k, l and t_{avg}.

Now, we study the influence of different parameters on the sensitivity of the sequential change point-detection algorithm. In all experiments, we set α, the parameter to control the degree a deviation from the reference distribution is considered as critical, to 0.95 and γ which controls the probability of false alarms to 0.1. Next, the parameter ν used in the detection of the end time of the alarm is set to 8 and M specifying the number of sampled sequences for computing a suitable threshold is set to 100.

We vary the time interval t_{avg} for averaging optical flow histograms as well as parameters controlling the estimation of the reference distribution based on previous observations $\{Y_j\}_{j=i-k-l}^{i-k}$ where l is the length of historical data used for estimating the reference distribution and k is a fixed time interval (see Figure 3.12). Figures 4.9(a) to 4.9(c) show results of the sequential change point detection algorithm for different values of k, l and t_{avg}.

In summary, we test 30 different combinations of these parameters and check if all three phases are detected (entering of pedestrians, congestion, increasing speeds at the end of the video). Table 4.2 shows precision, recall and F1 score for different parameters k, l and t_{avg}. Clearly, the results obtained for $t_{avg} = 75$ frames are better compared to $t_{avg} = 25$ and $t_{avg} = 50$, since sym_t values are smoother which can also be seen from Figure 4.9. For $t_{avg} = 75$ frames, the sequential change point detection correctly detects all change points for 8 of 10 parameter settings. From Table 4.2, it also becomes clear that the best results are obtained when using $l = 15s$ or $l = 20s$ for estimating the reference distribution.



(a) Time interval for averaging optical flow histograms: $t_{avg} = 25$ frames.

(b) Time interval for averaging optical flow histograms: $t_{avg} = 50$ frames.

(c) Time interval for averaging optical flow histograms: $t_{avg} = 75$ frames.

FIGURE 4.9: Sequential change-point detection on feature values over time using different time intervals for averaging optical flow histograms ((a): 25, (b): 50, (c): 75 frames) while studying the influence of the parameters k and l. Red regions indicate decreasing feature values, whereas green regions indicate increasing values.

In order to test the sensitivity of the feature with respect to the viewpoint, we generated synthetic videos from different viewpoints using the trajectories of the pedestrians extracted within the Hermes project. The camera positions are depicted in Figure 4.10(b). A grid is superimposed over the video frame and the $sym_{t,c}$ feature is calculated for each cell. Figure 4.10(a) shows the sum of $sym_{t,c}$ values of all cells over time for five different viewpoints. The different phases are clearly visible, although the change points are more prominent for viewpoints 3, 4 and 5.

Figure 4.10(c) shows alarms raised by the sequential change point detection algorithm (with $l = 10s, k = 5s, t_{avg} = 75$ frames) for all synthetic video sequences. The entrance of pedestrians is detected in all videos, whereas the congestion is not detected for viewpoint 1. The increase in velocities at the end is detected in three video sequences.

In our last series of experiments, we use the videos recorded from topview cameras (see Figures 4.7(a) and 4.7(b)) and detect phase changes in the movement of the pedestrians. Due to the overhead viewpoint, lateral body oscillation of pedestrians do not result in leftwards and rightwards motion. As a consequence, we do not make use of the original feature for congestion detection. However, from an overhead viewpoint, we can directly make use of the magnitude of optical flow motion vectors to estimate the velocity of the pedestrians. Thus, we adapt the feature for congestion detection by computing the center of mass of the two-dimensional optical flow histogram. Let $m_{dir,mag,t}$ denote the histogram value (or the mass) of the histogram bin with direction dir and magnitude mag at time t. The center of mass $(c_{dir,t}, c_{mag,t})$ of the histogram is computed as the average of the histogram bins weighted by their masses:

$$c_{dir,t} = \frac{\sum_{dir,mag} m_{dir,mag,t} \cdot dir}{\sum_{dir,mag} m_{dir,mag,t}} \tag{4.4}$$

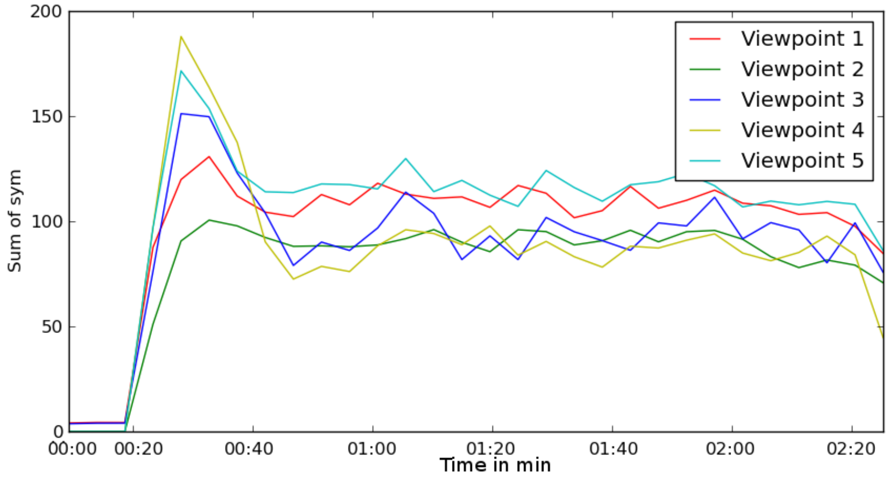
$$c_{mag,t} = \frac{\sum_{dir,mag} m_{dir,mag,t} \cdot mag}{\sum_{dir,mag} m_{dir,mag,t}} \tag{4.5}$$

which can be simplified to

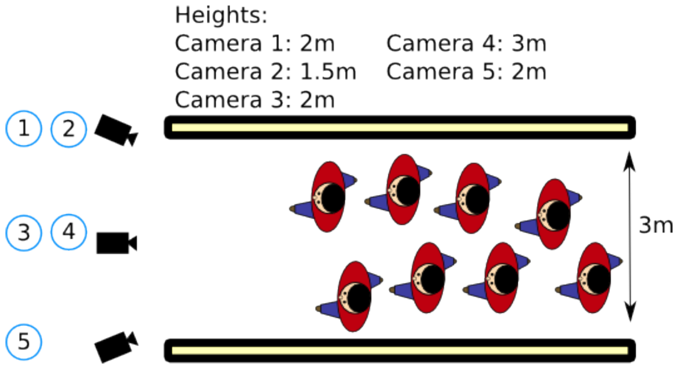
$$c_{dir,t} = \sum_{dir,mag} m_{dir,mag,t} \cdot dir \tag{4.6}$$

$$c_{mag,t} = \sum_{dir,mag} m_{dir,mag,t} \cdot mag \tag{4.7}$$

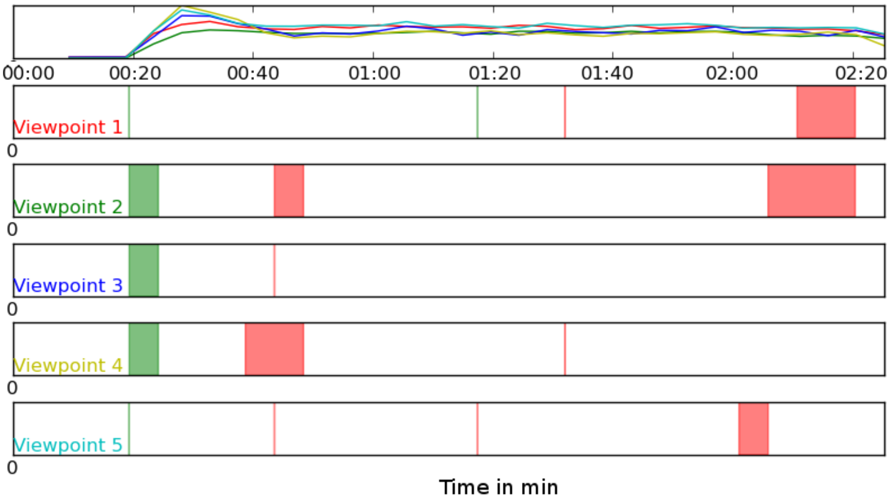
since the optical flow histogram is normalized and, hence, $\sum_{dir,mag} m_{dir,mag,t}$ sums up to 1. Now, we take $c_{mag,t}$ as a feature for congestion detection for topview cameras. Low values of $c_{mag,t}$ indicate that the velocity of the people is low which is indicative for a congestestion.



(a) Feature values over time for different viewpoints. Phase changes are clearly visible.

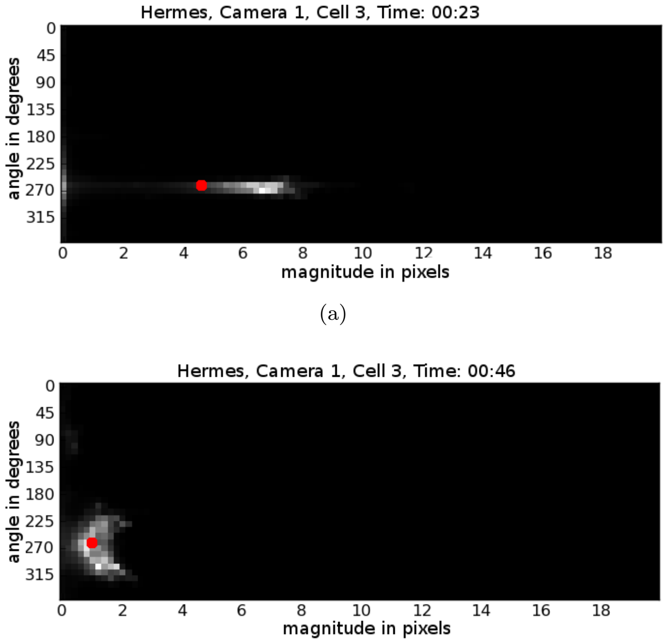


(b) Camera positions of synthetic video sequences.



(c) Alarms raised by the sequential change point detection algorithm. Red regions indicate decreasing feature values, whereas green regions indicate increasing values. For convenience, the feature values shown in Figure 4.10(a) are depicted again in the first row of Figure 4.10(c).

FIGURE 4.10: (a) Feature values and (c) raised alarms over time for different viewpoints. Trajectories extracted within the Hermes project are used to generate synthetic videos. Camera positions are shown in Figure (b).



(a)

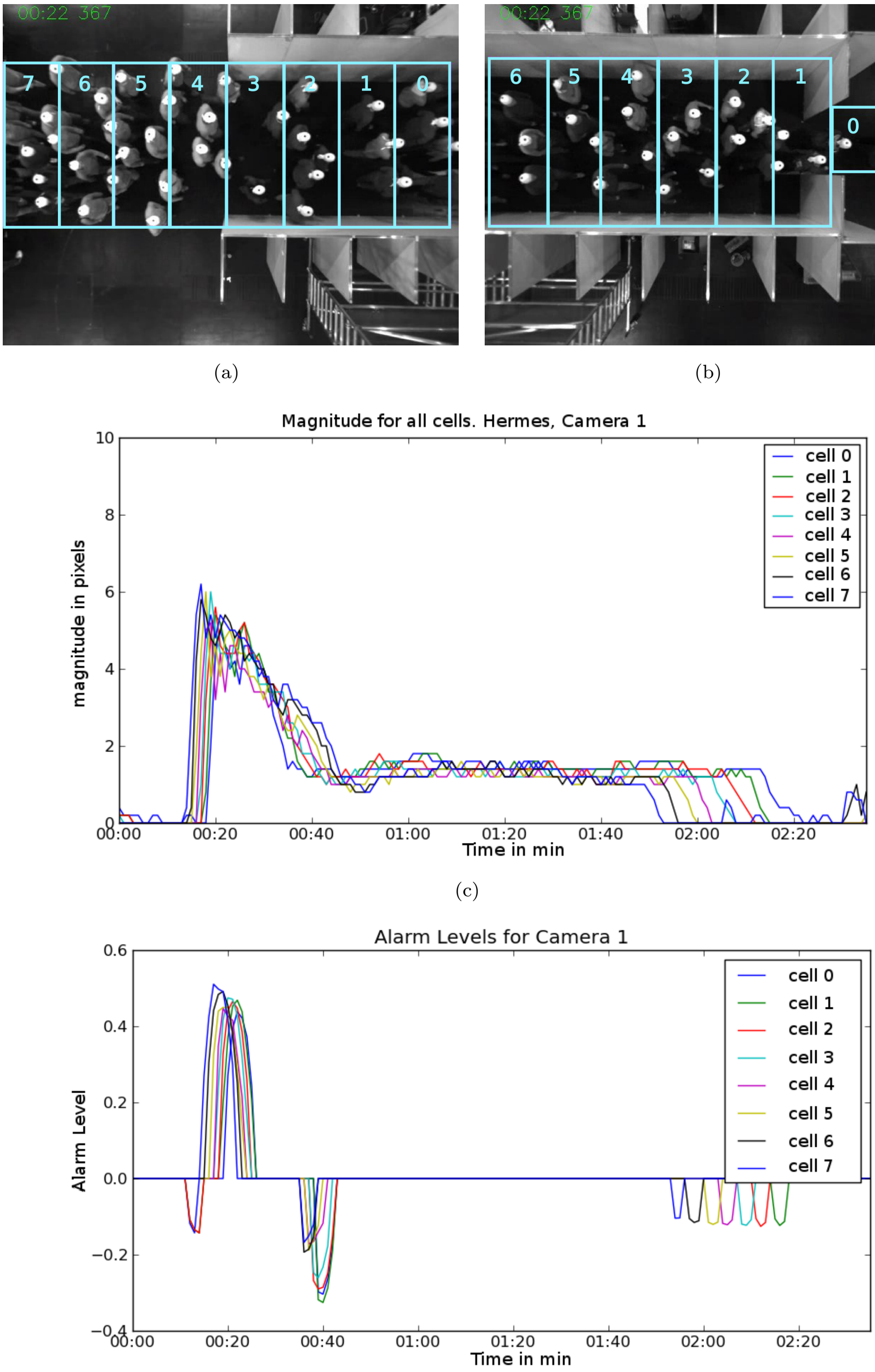
(b)

FIGURE 4.11: Two different optical flow histograms obtained from camera 1 of the Hermes dataset. In (a), people are moving fast, whereas in (b), they are going slowly. Instead of using the feature described in Section 3.3.1, we directly make use of the magnitude of optical flow motion vectors and compute the center of mass of the optical flow histogram which is marked in red here.

Figure 4.11 shows two optical flow histograms obtained from camera 1 (see Figure 4.7(a) for a screenshot). The center of mass is marked in red. The histogram in Figure 4.11(a) was computed at a time point, when people were moving fast. Here, $c_{mag,t}$ has a value of 4.5. Contrarily, the histogram in 4.11(b) corresponds to slow motion and $c_{mag,t} = 1.06$.

Figure 4.12(c) shows the development of $c_{mag,t}$ over time for all cells of camera 1. See Figure 4.12(a) for the locations of the cells. One can easily identify different phases in this plot. After approximately 18 seconds, people enter the camera's field of view. They are going quite fast which results in a high value of $c_{mag,t}$. This value decreases to a low value at time 00:40 min when the corridor is congested. This phase of low velocity lasts until 01:50 min (for cell 0) or 02:20 min (for cell 7), respectively, when people leave the field of view of camera 1. Note the time offset of the curves for different cells.

Now, we apply the algorithm described in section 3.3.2 to automatically detect change points in these curves. We set k to $10s$ and use $l = 10s$ for estimating a reference distribution. Figure 4.12(d) shows resulting alarm levels that describe the severity of an alarm. Here, alarm levels greater than zero indicate an increasing value of $c_{mag,t}$, whereas values less than zero indicate a decreasing signal. Our method successfully detects three change-points: At time 00:20 min, when people enter the field of view, next at 00:40 min, the beginning of the congestion and around 02:00 min, when people leave the camera's



(a) (b)

(c)

(d)

Figure 4.12: A grid of cells is placed onto the frame, where each cell covers an area of approximately 1m x 3m except for cell 0 in (b) which covers approximately 1m x 1.2m. Note that the fields of view of the cameras overlap: Cells 5 and 6 seen from camera 2 (Figure (b)) correspond to cells 0 and 1 observed by camera 1 (Figure (a)). The values of $c_{mag,t}$ over time for all cells (see Figure 4.12(a)) of camera 1 are shown in (c). One can easily identify different phases in this plot. Interestingly, the time offset of the curves for different cells is clearly visible. Figure (d) shows alarm levels reported by our change-point detection algorithm. Alarm levels greater than zero correspond to an increasing value of $c_{mag,t}$, lower values indicate a decreasing signal.

field of view. In particular, our method detects the phase of congestion and reports an alarm level of -0.3, which indicates a decreasing signal and thus is a sign of a congestion. Two false alarms are reported after 12 seconds in cell 0 and cell 2, because the system has not yet integrated enough observations. Note, that similar to Figure 4.12(c), the time offset between different cells is clearly visible.

We also apply the described procedure to video footage from camera 2 (see Figure 4.12(b) for a screenshot) with the same parameters as described previously. Our method successfully detects all three change-points without reporting any false alarm.

4.1.3 Subway Dataset

We applied the feature described in Section 3.3.1 to video footage consisting of six hours recorded in a subway station in Bonn, Germany, on the day of the *Rhein in Flammen* fireworks festival. The fireworks start at 23:15 and last until 23:36. See Figure 4.13 for an exemplary view.

We superimpose a grid of cells over the video frame as depicted in Figure 4.13. Next, we compute optical flow fields averaged over 2 minutes and compute feature values $sym_{t,c}$ for cell c at time t. Figure 4.14 shows sym_t, the sum over all cells at time t. Next, we apply the sequential change-point detection algorithm to sym_t (with $\alpha = 0.95$, $\gamma = 0.1$, $\nu = 8$, $M = 100$) and detect unusual events which are colored in red (decreasing signal) and green (increasing signal) in Figure 4.14. Here, we use observations from a window of 30 minutes to estimate a reference distribution and set $k = 15$ minutes.

Our system reports five alarms with the first one being a false alarm due to insufficient amount of data. Other alarms correspond to changing levels of pedestrian density. See Figure 4.15 for screenshots corresponding to these alarms. Most notably, the time point (b) shortly before the beginning of the fireworks is classified as an unusual event. Here, the number of people on the platform increases. After the fireworks, the number of people at the platform increases regularly as trains arrive. In the graph, this is reflected by peaks. Note that since the pedestrian density at the subway station never reaches a critical level, we do not observe lateral oscillations described in sections 2.2 and 3.2.3. Instead, people move rather fast resulting in high feature values in Figure 4.14.

4.1.4 Loveparade Dataset

We tested the approach to detect unusual events and congestions on video footage from camera 14 and 15 of the Loveparade dataset. Figures 4.16 and 4.18 show the development of the feature value measuring the mirror symmetry of the optical flow histograms. To

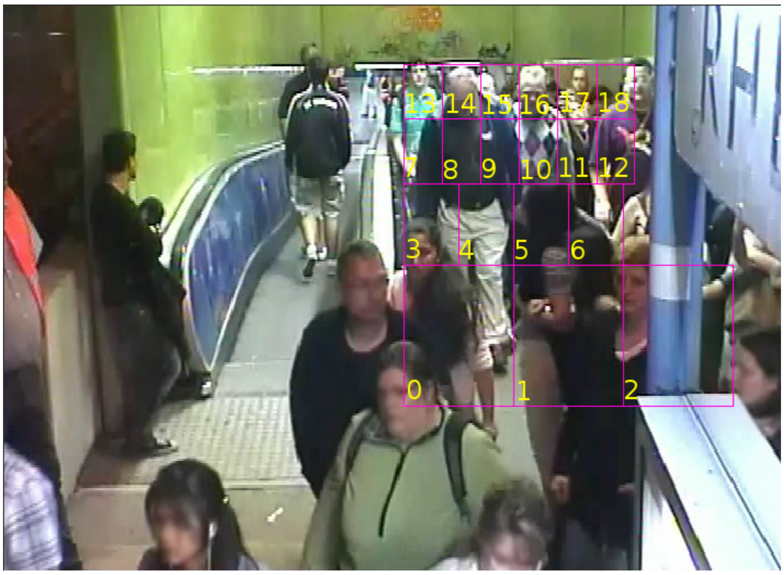


Figure 4.13: Exemplary screenshot of the subway dataset recorded on May 7, 2011 at the subway station *Rheinaue* in Bonn, Germany. A grid of cells is placed onto the video frame. Note that we blurred faces due to privacy concerns.

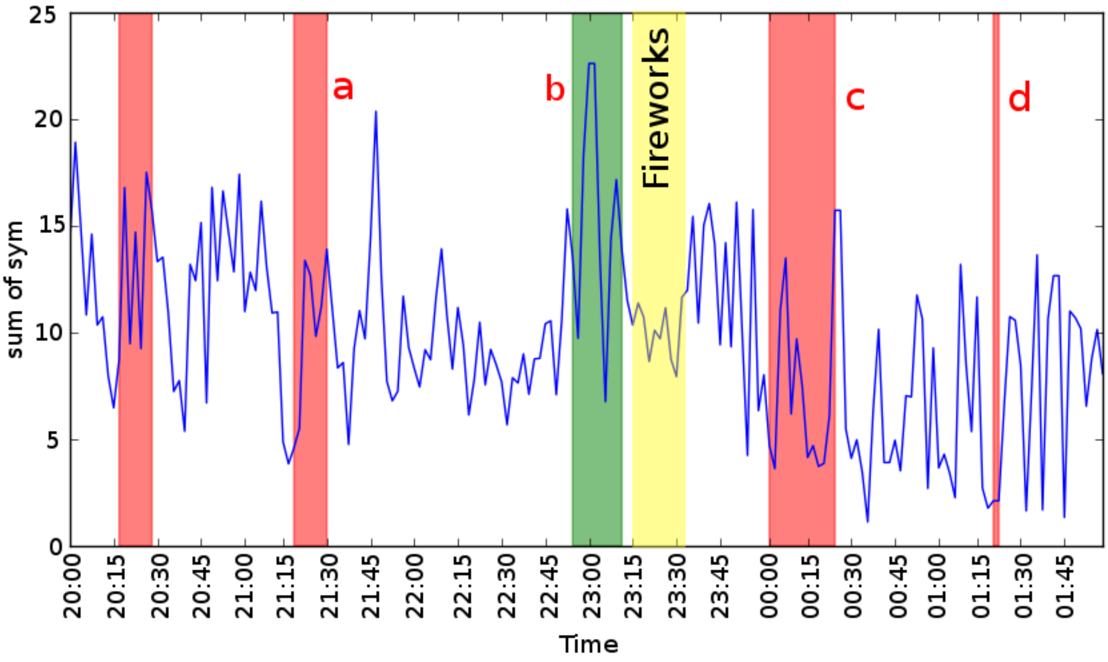


Figure 4.14: Development of the feature evaluated on the subway dataset and detected unusual events. Screenshots of the detected unusual events are depicted in Figure 4.15. The first alarm is a false alarm whereas the next alarms correspond to changing levels of pedestrian density. The increase of pedestrians (see Figure 4.15(b)) at the subway station shortly before the fireworks start is correctly identified. Peaks in the graph correspond to the arrival of trains.

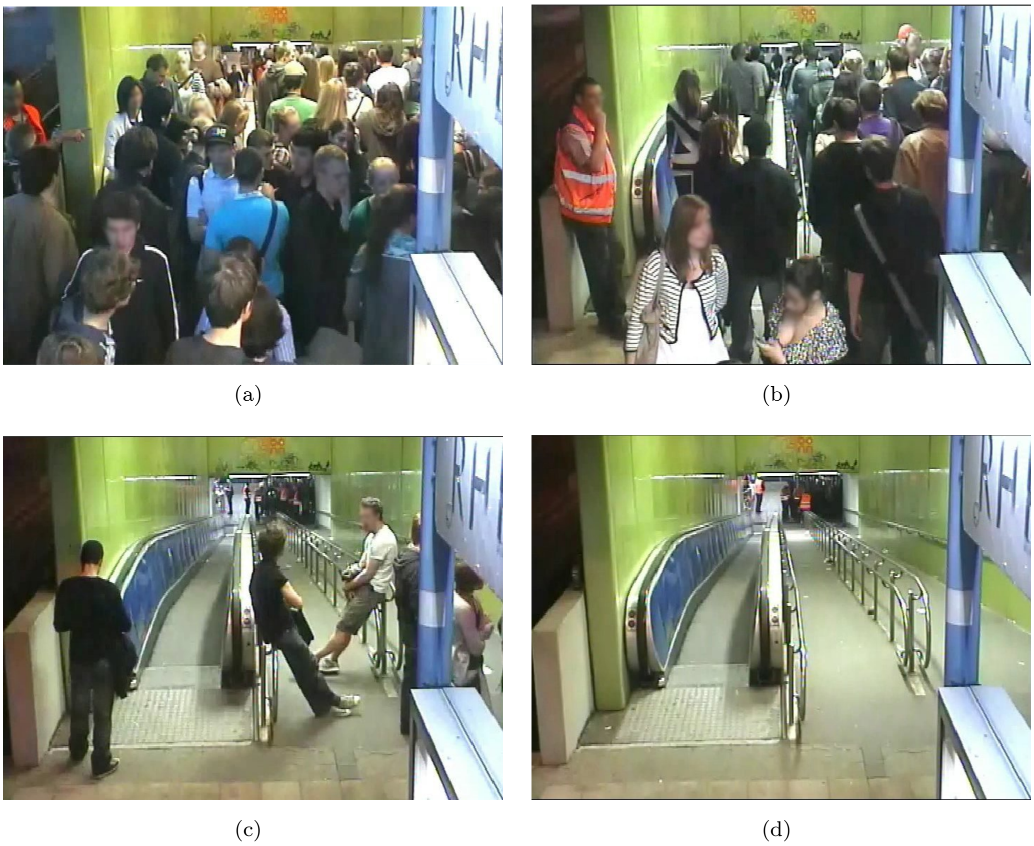


Figure 4.15: Detected unusual events in the subway dataset corresponding to different levels of density. See Figure 4.14 for time points of these events. Note that we blurred faces due to privacy concerns.

create these plots, we averaged histograms of optical flow over a time period of 10 seconds and summed the values of $sym_{t,c}$ for the cells in the scene foreground giving sym_t. We automatically detect change-points in an online manner using the cusum algorithm. In all experiments, we set α, the parameter to control the degree a deviation from the reference distribution is considered as critical, to 0.95 and γ which controls the probability of false alarms to 0.1. Next, the parameter ν used in the detection of the end time of the alarm is set to 8 and M specifying the number of sampled sequences for computing a suitable threshold is set to 100. We perform multiple runs with different parameters for k and l. In the lower part of Figures 4.16 and 4.18, we show obtained results for each run for both cameras. Alarms that have been raised by our system are colored in red, if the signal is at a low level, whereas a jump to a high value is marked in green.

Comparing the automatically detected alarms with the video footage reveals that most of the raised alarms correspond to anomalies in the video, e.g. ambulances or police cars crossing the scene. Figures 4.17 and 4.19 depict screenshots corresponding to the raised alarns for camera 14 and 15. In particular, at approx. 16:22 the system raises a severe alarm ($L > 0.8$) and reports low values of sym_t which is indicative for a congestion. In fact, at this time point the crowd is densely packed and has come to a halt, see Figures 4.17(i) and 4.19(k). In this situation, our system would have detected a very critical situation and alarmed the security personnel to take necessary actions in order to prevent a deadly stampede.

A few false alarms at the beginning of the video recordings are reported, since our system has not yet integrated enough observations of normal crowd behavior. Only the run with $k = 5$ min and $l = 5$ min produces too many false alarms.

In summary, our system detects major unusual events such as the congestion, the police cordon and crossing of cars in nearly all parameter settings. Although the proposed feature is designed for being sensitive to lateral oscillations in order to detect congestions, our experiments show that it is also sensitive to changes in pedestrian flow caused by crossing cars or closed access control systems, for example.

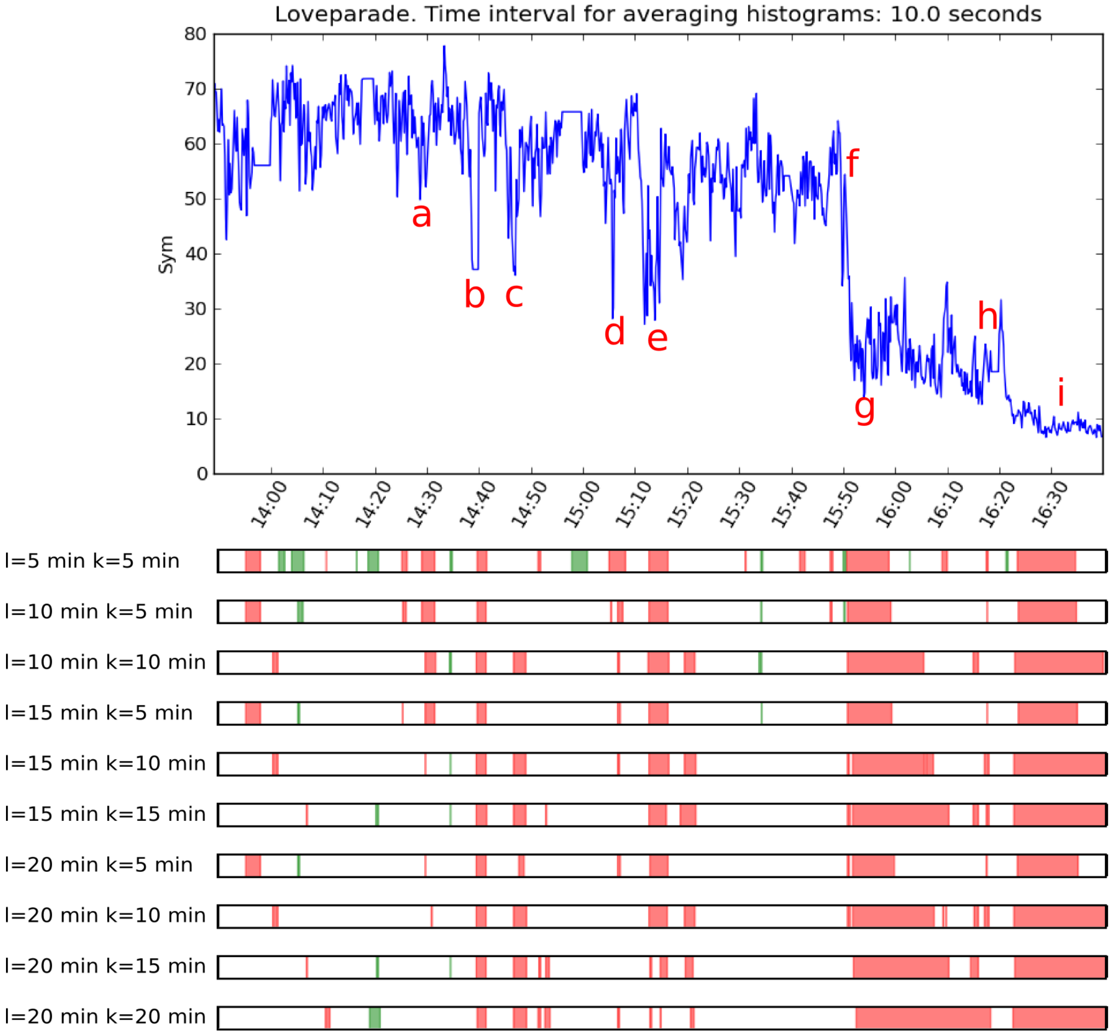
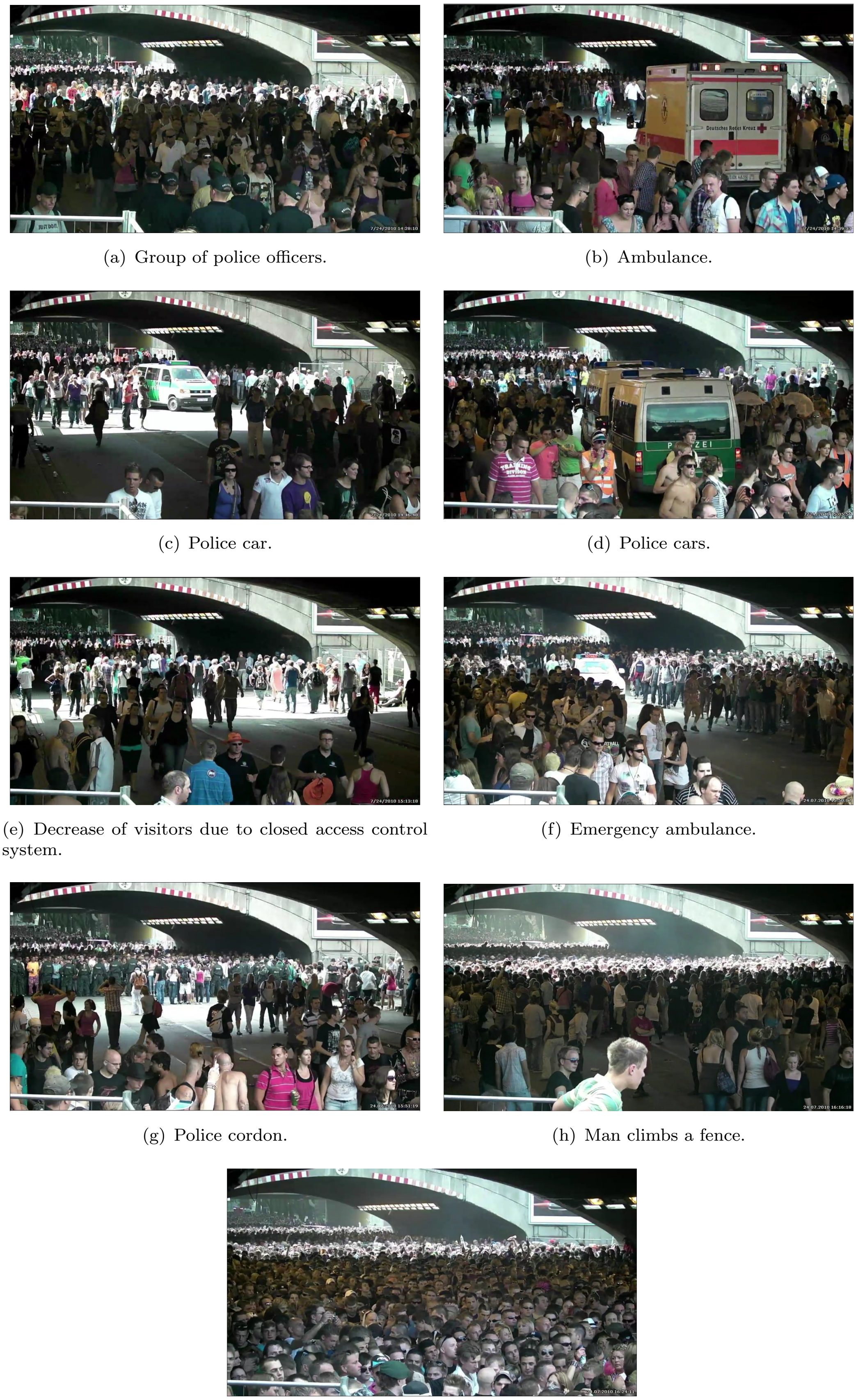


FIGURE 4.16: Detection results on videos of camera 15. Development of the feature measuring the mirror symmetry of the optical flow histograms. One can clearly distinguish phases of different pedestrian flow as well as unusual events. By applying a sequential change-point detection algorithm, unusual events and congestions in particular can be detected automatically. Detection results are depicted in the lower part of the figure where we use different parameters for k and l. Red regions indicate decreasing feature values, whereas green regions indicate increasing values. Unusual events that have been reported by the system for most of the parameter combinations are marked in the graph. The corresponding screenshots are shown in Figure 4.17. We detect unusual events such as police cars crossing the scene, changing pedestrian flow due to closed access control systems and a congestion starting at 16:22 which is characterized by extremely low feature values.



(a) Group of police officers.

(b) Ambulance.

(c) Police car.

(d) Police cars.

(e) Decrease of visitors due to closed access control system.

(f) Emergency ambulance.

(g) Police cordon.

(h) Man climbs a fence.

(i) Congestion.

Figure 4.17: Detected unusual events from camera 15 of the Loveparade dataset. See Figure 4.16 for time points of these events.

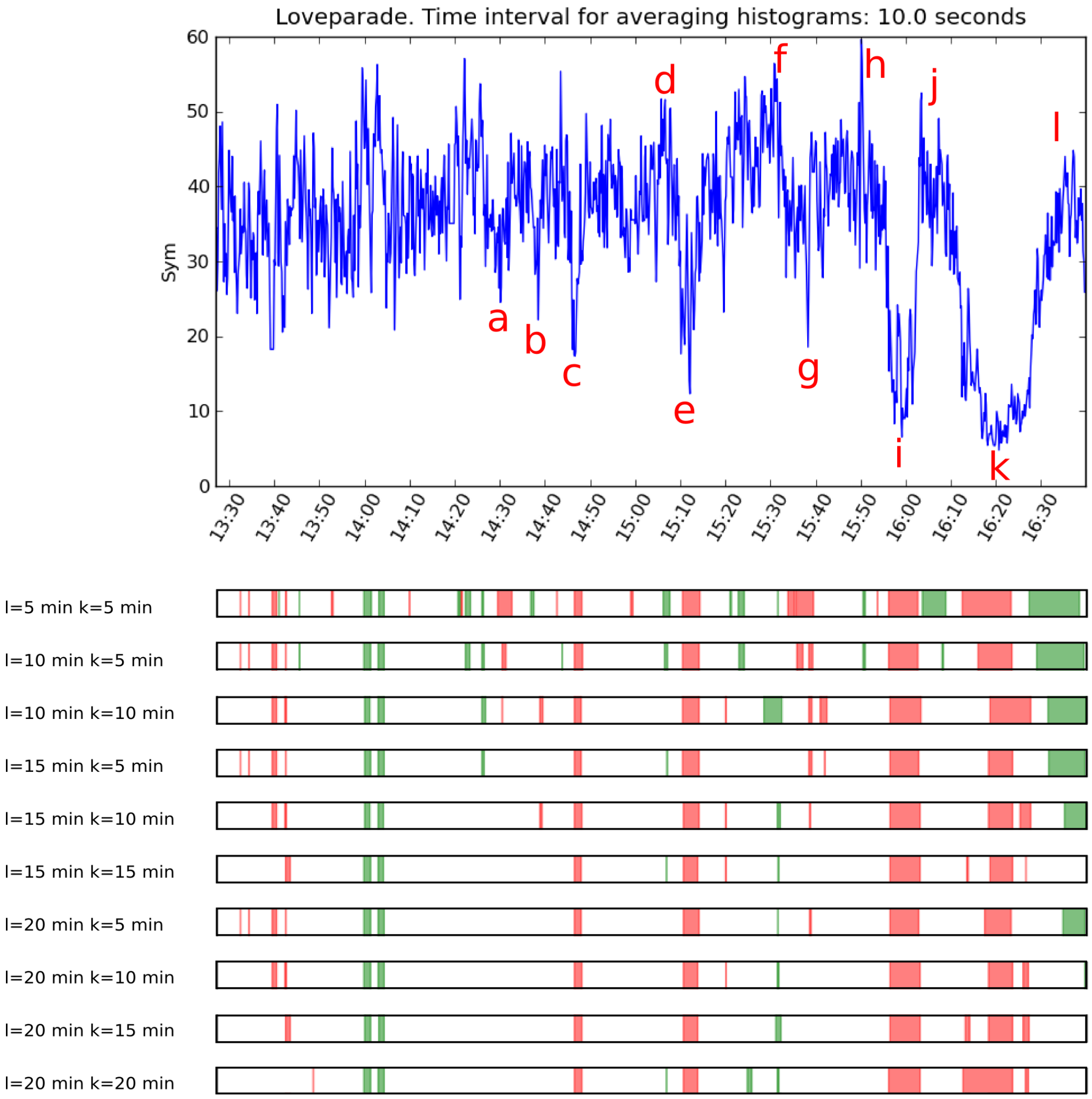
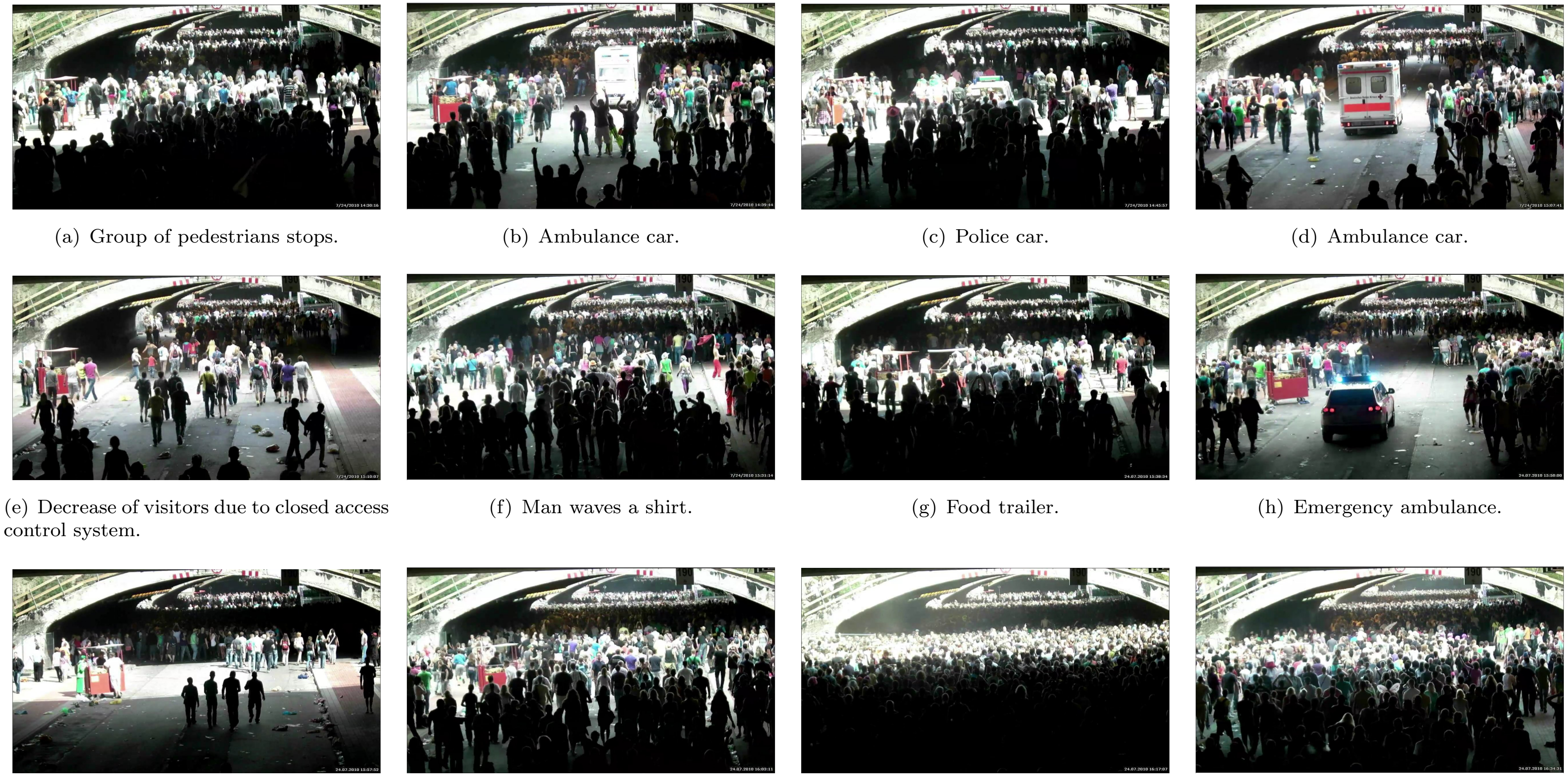


FIGURE 4.18: Detection results on videos of camera 14. Development of the feature measuring the mirror symmetry of the optical flow histograms. One can clearly distinguish phases of different pedestrian flow as well as unusual events. By applying a sequential change-point detection algorithm, unusual events and congestions can be detected automatically. Detection results are depicted in the lower part of the figure where we use different parameters for k and l. Red regions indicate decreasing feature values, whereas green regions indicate increasing values. Unusual events that have been reported by the system for most of the parameter combinations are marked in the graph. The corresponding screenshots are shown in Figure 4.19. We detect unusual events such as police cars crossing the scene, changing pedestrian flow due to closed access control systems and a congestion characterized by extremely low feature values.



(a) Group of pedestrians stops.

(b) Ambulance car.

(c) Police car.

(d) Ambulance car.

(e) Decrease of visitors due to closed access control system.

(f) Man waves a shirt.

(g) Food trailer.

(h) Emergency ambulance.

(i) Decrease of visitors due to closed access control system.

(j) Increase of visitors due to re-opened access control system.

(k) Congestion.

(l) Dissolving of congestion.

Figure 4.19: Detected unusual events from camera 14 of the Loveparade dataset. See Figure 4.18 for time points of these events.

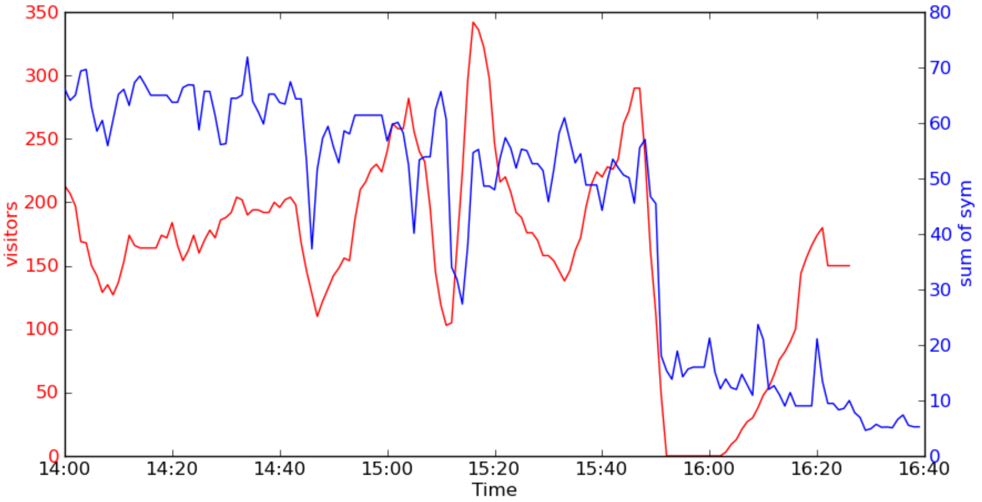


FIGURE 4.20: Comparison of the proposed feature (blue) evaluated from video recordings of camera 15 to the number of visitors (red) obtained in a study of the pedestrian streams at the Loveparade [97]. Although our feature is not designed for counting people, it gives a rough estimate of this quantity as the highly correlated curves show.

A comparison with a study of pedestrian streams at the Loveparade shows that our feature is also highly correlated with the number of visitors. This study of the pedestrian streams at the Loveparade [97] has been conducted within the BMBF-funded project EVA [98]. Based on video recordings, visitors were counted at the main station of Duisburg, in the western bridge area and the eastern tunnel. The number of visitors (per minute) in the western bridge area is depicted in red in Figure 4.20. Our feature evaluated from video recordings of camera 15 is depicted in blue in the same Figure. In contrast to previous experiments, we averaged optical flow histograms over a time interval of one minute in order to compare both quantities. Although our system is not able to count the number of people directly, the two curves are highly correlated which shows that our system not only detects unusual events, but also gives a rough estimate of the development of visitor attendance.

4.2 Detection of Crowd Turbulences

Due to the lack of data showing crowd turbulences, it is rather difficult to evaluate methods for detecting shock waves. However, we tested our approach (see Section 3.4) on videos obtained from camera 13 of the Loveparade dataset which monitored the main entry ramp to the festival area. The video shows short sequences of shock waves propagating through the densely packed crowd. Regions of high pressure are automatically detected by our method and are depicted in Figure 4.21. Two cells at the border of the frame are correctly classified as alarming cells showing a shock wave. In total, we extracted approximately 4 minutes of video sequences of camera 13 showing a densely packed crowd. Three shock waves can be observed which are correctly detected by our method. However, our method also reports two false alarms.

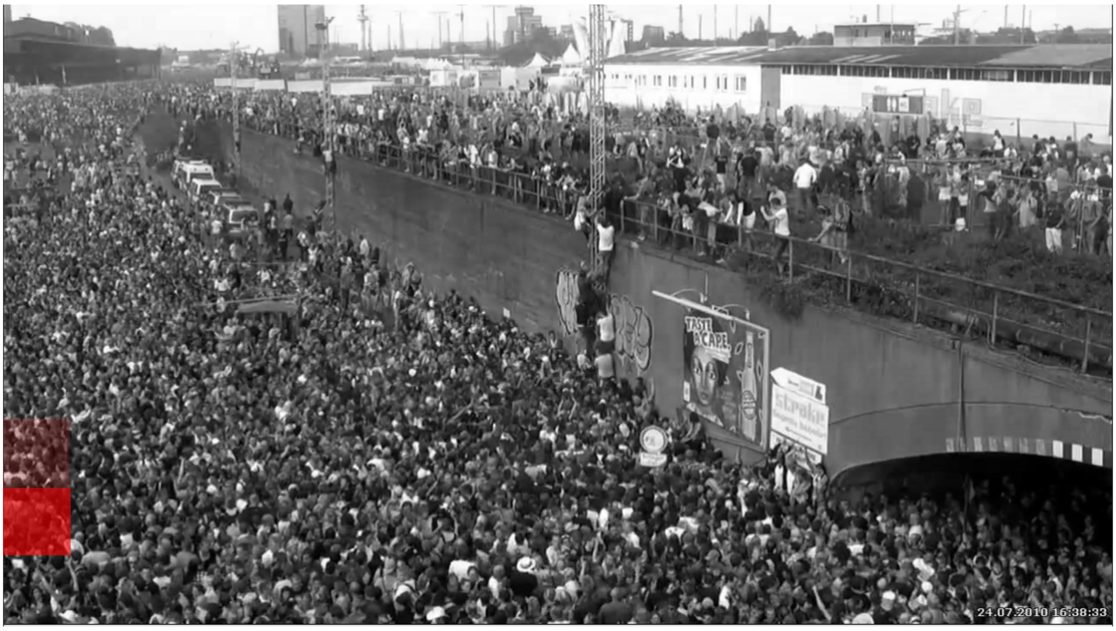


FIGURE 4.21: Detection of Crowd Turbulences. Camera 13 which monitored the main entry ramp shows short sequences of shock waves that are propagated. We automatically detect regions of high pressure which are colored red in the above Figure. Note that the saturation of the alarming cells have been increased for the sake of visibility.

Chapter 5

Simulation of Pedestrian Motion in Crowds

Traditionally, pedestrian simulations are a standard tool in public space design. Urban infrastructure as well as buildings are planned with the support of simulation studies. They are used for guiding pedestrian streams while avoiding obstacles or for optimizing routes to platforms in train stations, for example. Optimal locations of facilities such as access control systems, information desks, stairways, elevators etc. are identified or the dimensions of sidewalks, doors, or passages are determined. In crowd management, dangerous locations are identified and defused in order to prevent accidents. When developing evacuation strategies, simulations are conducted for optimizing evacuation routes as well as dimensions, quantity and locations of doors. In particular, when minimizing evacuation times or identifiying hazardous locations, it is of vital importance that simulations are as accurate and realistic as possible.

Considering future research directions that integrate real-time information into simulations in order to foresee hazardous situations, prevent accidents and recommend evacuation strategies in real-time [99], accurate pedestrian models are even more important. Criterions for the quality of models for pedestrian motion are the capability of the model to reproduce the fundamental diagram (Section 2.2.1) as well as empirical observations such as lane formation or stop-and-go waves (Sections 2.2.2 and 2.3).

5.1 The Generalized Centrifugal Force Model

The most popular physical model for pedestrian simulations is the social force model presented by Helbing et al. [12, 21]. Despite of many modifications of the original model that have been proposed over the years [22–28], the model still cannot reproduce realistic motion behavior in areas of high pedestrian densities. Overlapping of pedestrians can hardly be avoided making a higher repulsive force between pedestrians necessary. As a consequence, oscillations occur when high repulsive forces push pedestrians back and forth. In their *centrifugal force model*, Yu et al. [26] take the relative velocity of people into account and introduce a collision detection technique in order to avoid overlappings. However, by introducing a collision detection procedure, the centrifugal force model as proposed in [26] increases the complexity of the model and counteracts the idea of a force-based system. Chraibi and Seyfried [27, 100] propose the *generalized centrifugal force model* (GCFM) in which they replace the collision detection technique in favor of an improved formulation of repulsive forces. Below, we follow the notation of Chraibi and Seyfried [27].

Given pedestrian i with coordinates $\overrightarrow{R_i}$ and mass m_i, the movement of i is described as the superposition of a driving force and repulsive forces:

$$m_i \ddot{\overrightarrow{R_i}} = \overrightarrow{F_i^{drv}} + \sum_{j \in \mathcal{N}_i} \overrightarrow{F_{ij}^{rep}} + \sum_{w \in \mathcal{W}_i} \overrightarrow{F_{iw}^{rep}}. \tag{5.1}$$

The driving force $\overrightarrow{F_i^{drv}}$ models the desired motion direction of pedestrian i. Other pedestrians $j \in \mathcal{N}_i$ in the neighborhood of i induce a repulsive force $\overrightarrow{F_{ij}^{rep}}$ acting on pedestrian i in order to avoid collisions. The set of nearby pedestrians $\mathcal{N}_i$ is defined as

$$\mathcal{N}_i = \{j : \|\overrightarrow{R_j} - \overrightarrow{R_i}\| \leq r_c \wedge i \text{ “feels” } j\}. \tag{5.2}$$

Here, r_c is a cutoff radius and i feels j if the view from i to j is unobstructed. Similarly, obstacles $w \in \mathcal{W}_i$ with

$$\mathcal{W}_i = \{w : \|\overrightarrow{R_{wi}} - \overrightarrow{R_i}\| \leq r_c\} \tag{5.3}$$

cause a repulsive force influencing the motion of pedestrian i.

Using numerical integration of the resulting system of differential equations, the positions and velocities of the pedestrians are calculated at each point in time.

Driving Force

The driving force models motion into the direction of the pedestrian's intended destination as well as his desired speed v_i^0. Formally, the driving force of pedestrian i is given as

$$\overrightarrow{F_i^{drv}} = m_i \frac{\overrightarrow{v_i^0} - \overrightarrow{v_i}}{\tau} \tag{5.4}$$

where $\overrightarrow{v_i}$ denotes i's current velocity and τ is a relaxation time.

Repulsive Force

Repulsive forces induced by nearby pedestrians as well as obstacles model the avoidance of collisions. In [27], the repulsive force between pedestrians i and j is given as

$$\overrightarrow{F_{ij}^{rep}} = -m_i k_{ij} \frac{(\eta_{ped} \|\overrightarrow{v_i^0}\| + v_{ij})^2}{dist_{ij}} \overrightarrow{e_{ij}}. \tag{5.5}$$

The model is named *centrifugal force model*, since the repulsive force in Equation 5.5 is similar to the centrifugal force in mechanics.

With $\overrightarrow{R_{ij}}$ being the vector pointing from pedestrian i to pedestrian j, $\overrightarrow{e_{ij}}$ is the normalized connecting vector (see Figure 5.1):

$$\overrightarrow{R_{ij}} = \overrightarrow{R_j} - \overrightarrow{R_i}, \ \overrightarrow{e_{ij}} = \frac{\overrightarrow{R_{ij}}}{\|\overrightarrow{R_{ij}}\|} \tag{5.6}$$

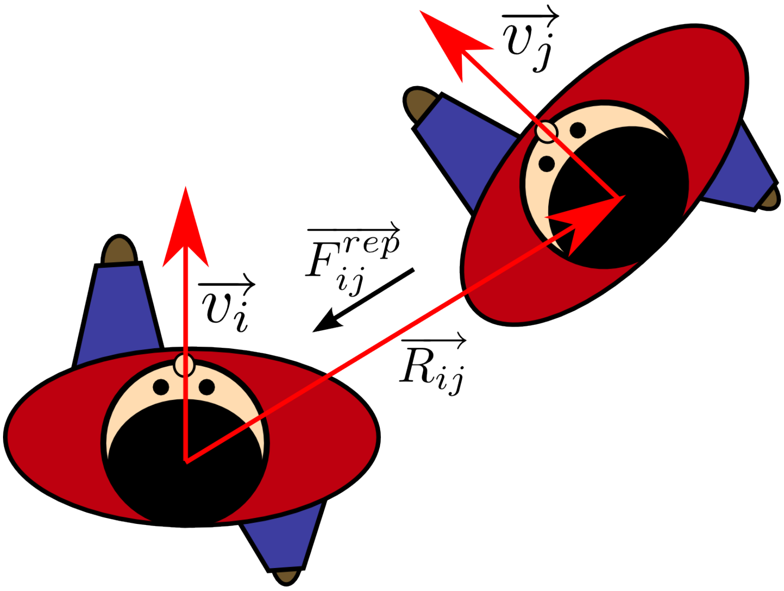


FIGURE 5.1: Repulsive force acting on pedestrian i walking with velocity $\overrightarrow{v_i}$. The force is exerted by pedestrian j walking with velocity $\overrightarrow{v_j}$. $\overrightarrow{R_{ij}}$ is the vector pointing from i to j.

In order to model the field of perception, the coefficient k_{ij} in Equation 5.5 is at its maximum when pedestrian j is in the motion direction of pedestrian i, whereas it is zero when pedestrian j is out of sight, i.e. when the angle between $\overrightarrow{v_i}$ and pedestrian j is greater or equal to 90°:

$$k_{ij} = \begin{cases} (\overrightarrow{v_i} \cdot \overrightarrow{e_{ij}}) / \|\overrightarrow{v_i}\|, & \text{if } \overrightarrow{v_i} \cdot \overrightarrow{e_{ij}} > 0 \wedge \|\overrightarrow{v_i}\| \neq 0 \\ 0, & \text{otherwise.} \end{cases} \tag{5.7}$$

In addition to the field of perception, the relative velocity between pedestrians is also taken into account. If pedestrians in front of pedestrian i are walking fast, i is not affected by them:

$$v_{ij} = \begin{cases} (\overrightarrow{v_i} - \overrightarrow{v_j}) \cdot \overrightarrow{e_{ij}}, & \text{if } (\overrightarrow{v_i} - \overrightarrow{v_j}) \cdot \overrightarrow{e_{ij}} > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{5.8}$$

However, if v_{ij} is small, the repulsive force decreases leading to overlappings. Chraibi and Seyfried take the intended speed $\|\overrightarrow{v_i^0}\|$ into account, since pedestrians with a high desired speed require higher repulsive forces in order to prevent collisions. They introduce a parameter η_{ped} to control the influence of the desired speed. In their experiments, they find a value of $\eta_{ped} = 0.3$ in order to minimize both overlappings and oscillations.

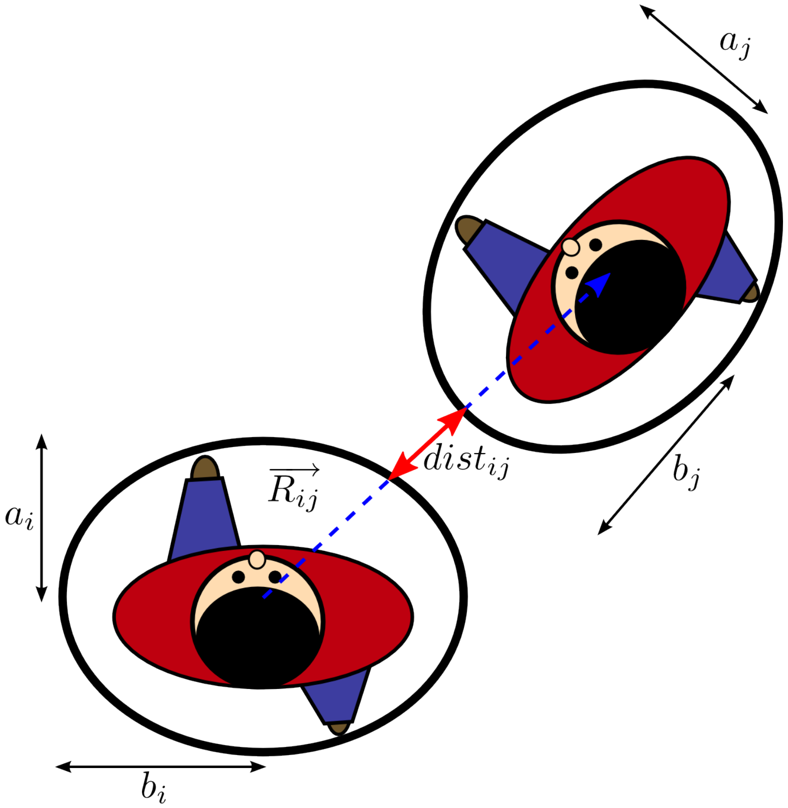


FIGURE 5.2: Pedestrians are modeled as ellipses with semi-axis a and b. $dist_{ij}$ is the distance between ellipses along the vector $\overrightarrow{R_{ij}}$ which connects the ellipse centers.

Next, $dist_{ij}$ in Equation 5.5 denotes the distance between pedestrians i and j which are modeled as ellipses. $dist_{ij}$ is the distance of the ellipses along the vector $\overrightarrow{R_{ij}}$ connecting the ellipse centers, see Figure 5.2.

Pedestrian i is modeled as an ellipse centered on $\overrightarrow{R_i}$ with semi-axis a_i being the semi-axis in the movement direction $\overrightarrow{v_i}$ and b_i the semi-axis orthogonal to a_i. a_i includes physical body extensions of pedestrian i as well as space requirements in movement direction which depends on the current speed:

$$a_i = a_{min} + \tau_a \|\overrightarrow{v_i}\| \tag{5.9}$$

where a_{min} and τ_a are two parameters to be set.

With b_i, Chraibi and Seyfried model space requirements into lateral direction due to lateral swaying. From observations of trajectories, they model b_i as:

$$b_i = b_{max} - (b_{max} - b_{min}) \frac{\|\overrightarrow{v_i}\|}{\|\overrightarrow{v_i^0}\|} \tag{5.10}$$

where b_{max} is the maximum amplitude of lateral swaying which decreases to a minimum of b_{min}.

Similar to the repulsive force between pedestrians, the repulsive force exerted by the wall w is given in [27] as:

$$\overrightarrow{F_{iw}^{rep}} = -m_i k_{iw} \frac{(\eta_{wall} \|\overrightarrow{v_i^0}\| + \|\overrightarrow{v_i^n}\|)^2}{dist_{iw}} \overrightarrow{e_{iw}} \tag{5.11}$$

where $\overrightarrow{v_i^n}$ is the normal component of i's velocity vector to the wall and η_{wall} is a parameter controlling the influence of the intended speed on the repulsive force similar to η_{ped} in Equation 5.5. $dist_{iw_j}$ is the distance between the nearest point on wall w and the nearest point on the ellipse of pedestrian i. $\overrightarrow{e_{iw}}$ and k_{iw} are defined analogously to Equations 5.6 and 5.7.

5.2 Integration of Characteristic Human Motion Patterns

In contrast to the original social force model, the generalized centrifugal force model shows good results even in areas of high pedestrian density. However, a comparison of simulated trajectories and real trajectories reveals that lateral swaying has not been adequately taken into account. In the generalized centrifugal force model, pedestrians are modeled as ellipses which include space requirements into lateral direction due to lateral swaying, but pedestrians do not actually perform lateral oscillations. Instead, they are walking along a straight line.

Since the observation of lateral swaying is a fundamental characteristic of human gait and crucial for the detection of congested areas as presented in Section 3.3, we introduce an *oscillation force*. Thus, the movement of a pedestrian i with mass m_i and position $\overrightarrow{R}_i$ is modeled as the superposition of a driving force, repulsive forces as well as the oscillation force:

$$m_i\ddot{\overrightarrow{R_i}} = \overrightarrow{F_i^{drv}} + \overrightarrow{F_i^{osc}} + \sum_{j \in \mathcal{N}_i} \overrightarrow{F_{ij}^{rep}} + \sum_{w \in \mathcal{W}_i} \overrightarrow{F_{iw}^{rep}}. \tag{5.12}$$

where $\overrightarrow{F_i^{drv}}$, $\overrightarrow{F_{ij}^{rep}}$ and $\overrightarrow{F_{iw}^{rep}}$ are defined as described above and $\overrightarrow{F_i^{osc}}$ is defined as:

$$\overrightarrow{F_i^{osc}} = -m_i(2\pi f(\overrightarrow{v_i}))^2 s(\overrightarrow{v_i}) \sin(2\pi f(\overrightarrow{v_i})t + \phi_0)\overrightarrow{n_i}. \tag{5.13}$$

Here, $\overrightarrow{n_i}$ is the unit vector normal to the moving direction $\overrightarrow{v_i}$ and ϕ_0 is the phase of oscillation. $f(\overrightarrow{v_i})$ and $s(\overrightarrow{v_i})$ are the frequency and amplitude of lateral swaying where we use the observations from Section 2.2.2 (Equations 2.1 and 2.2):

$$f(\overrightarrow{v_i}) = 0.44\|\overrightarrow{v_i}\| + 0.35 \tag{5.14}$$

and

$$s(\overrightarrow{v_i}) = -0.14\|\overrightarrow{v_i}\| + 0.21. \tag{5.15}$$

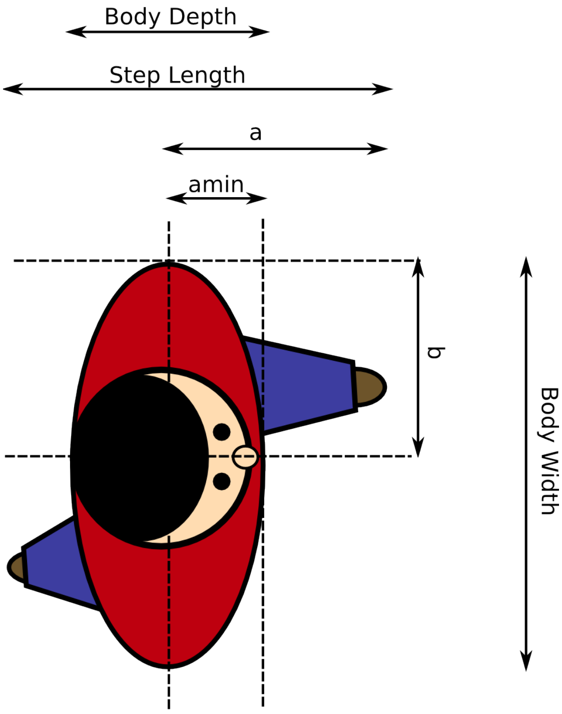


FIGURE 5.3: A Pedestrian is modeled as an ellipse with semi-axis a_i being the semi-axis into the motion direction and b_i the orthogonal semi-axis. Note that b_i is half of the body width and a_i is half of the step length.

In addition to the introduction of an oscillation force, we adapt the ellipse modeling space requirements of pedestrians. Similar to Chraibi and Seyfried, we model pedestrian i as an ellipse with semi-axes a_i and b_i (Figure 5.3) and take empirical studies of human body dimensions into account. Weidmann [29] reports a mean value of 0.23 m for the body depth with the 97.5 percentile at 0.27 m. Next, according to Weidmann, the step length can be computed as:

$$l(\overrightarrow{v_i}) = 0.235 + 0.302 \cdot \|\overrightarrow{v_i}\|. \tag{5.16}$$

First, we note that the first term in this equation is almost equal to the body depth of 0.23 m given above. The second term in Equation 5.16 models space requirements of a pedestrian for taking a step. Having in mind that a_i denotes the length of the semi-axis and thus is half of the step length, we set it as follows:

$$a_i(\overrightarrow{v_i}) = 0.5 \cdot l(\overrightarrow{v_i}) = a_{min} + a_\tau \cdot \|\overrightarrow{v_i}\| \tag{5.17}$$

with $a_{min} \in \mathcal{N}(0.5 \cdot 0.23, 0.01) = \mathcal{N}(0.115, 0.01)$ modeling half of the body depth and $a_\tau \in \mathcal{N}(0.5 \cdot 0.302, 0.001) = \mathcal{N}(0.151, 0.001)$ modeling space requirements for taking a step.

The second semi-axis b_i is set to $b_i \in \mathcal{N}(0.5 \cdot 0.46, 0.01) = \mathcal{N}(0.23, 0.01)$, since Weidmann gives a value of 0.46 m for the body width with the 97.5 percentile at 0.5 m. Note that in contrast to the generalized centrifugal force model, we do not take lateral space requirements into consideration here, since they are already modeled by the oscillation force given in Equation 5.13.

5.3 Experiments

Since lateral oscillations play an important role in areas of high pedestrian density and congestions in particular, we modified the generalized centrifugal force model to reproduce these characteristic human motion patterns. For that purpose, we superimposed an oscillation force and adapted the ellipse of pedestrians modeling their space requirements. In Section 5.3.1, we evaluate the model proposed in Section 5.2 and use it for simulating different critical situations in Section 5.3.2. Using the simulated trajectories, we generate synthetic videos and test our video analysis approach for detecting dangerous situations in crowds.

5.3.1 Simulation Results

In order to evaluate the quality of a model for simulating pedestrian behavior, one can test quantitative as well as qualitative criterions. To quantitatively evaluate the

Parameter	Description	Value
r_c	cutoff radius	2
v_i^0	desired speed	$\mathcal{N}(1.34, 0.26)$
m_i	mass	1
τ	relaxation time	$\mathcal{N}(0.5, 0.001)$
η_{ped}	controls influence of intended speed on repulsive force	0.3
η_{wall}	controls influence of intended speed on repulsive force	0.2
δt	step size for solving differential equation system	0.001
GCFM [27]		
a_{min}	minimum length of semi-axis	$\mathcal{N}(0.2, 0.01)$
τ_a	factor for computing length of semi-axis	$\mathcal{N}(0.53, 0.001)$
b_{max}	maximum length of semi-axis	$\mathcal{N}(0.25, 0.001)$
b_{min}	minimum length of semi-axis	$\mathcal{N}(0.2, 0.001)$
modified GCFM (Section 5.2)		
a_{min}	minimum length of semi-axis	$\mathcal{N}(0.115, 0.01)$
a_τ	factor for computing length of semi-axis	$\mathcal{N}(0.151, 0.001)$
b_i	length of semi-axis	$\mathcal{N}(0.23, 0.01)$

Table 5.1: Parameter values used in the experiments.

effectiveness of our model, we simulate pedestrian movements and test if the fundamental diagram is well reproduced. Secondly, we employ trajectories obtained from the Hermes dataset, use their initial positions and simulate pedestrian movements. A comparison of real trajectories to the simulated trajectories reveals that our model simulates realistic trajectories.

In our experiments, we use the parameters shown in Table 5.1.

For verification of the model proposed in Section 5.2 and to examine the influence of the two modifications (oscillation force and adaptation of ellipse), we simulate pedestrian trajectories in a corridor (26m × 1m) and measure the fundamental diagram in a measurement area (2m × 1m) located in the middle of the corridor.

The mean speed of pedestrian i that enters the measurement area at time t_i^{in} at location (x_i^{in}, y_i^{in}) and leaves it at time t_i^{out} at location (x_i^{out}, y_i^{out}) is computed as [27]:

$$v_i = \frac{\sqrt{(x_i^{out} - x_i^{in})^2 + (y_i^{out} - y_i^{in})^2}}{t_i^{out} - t_i^{in}}. \tag{5.18}$$

The density is then defined as:

$$\rho_i = \frac{1}{t_i^{out} - t_i^{in}} \int_{t_i^{in}}^{t_i^{out}} \rho(t) dt \tag{5.19}$$

with

$$\rho(t) = \frac{N_{in}(t)}{A_m} \tag{5.20}$$

where A_m is the area of the measurement area and N_{in} is the number of pedestrians present in the measurement area at time t.

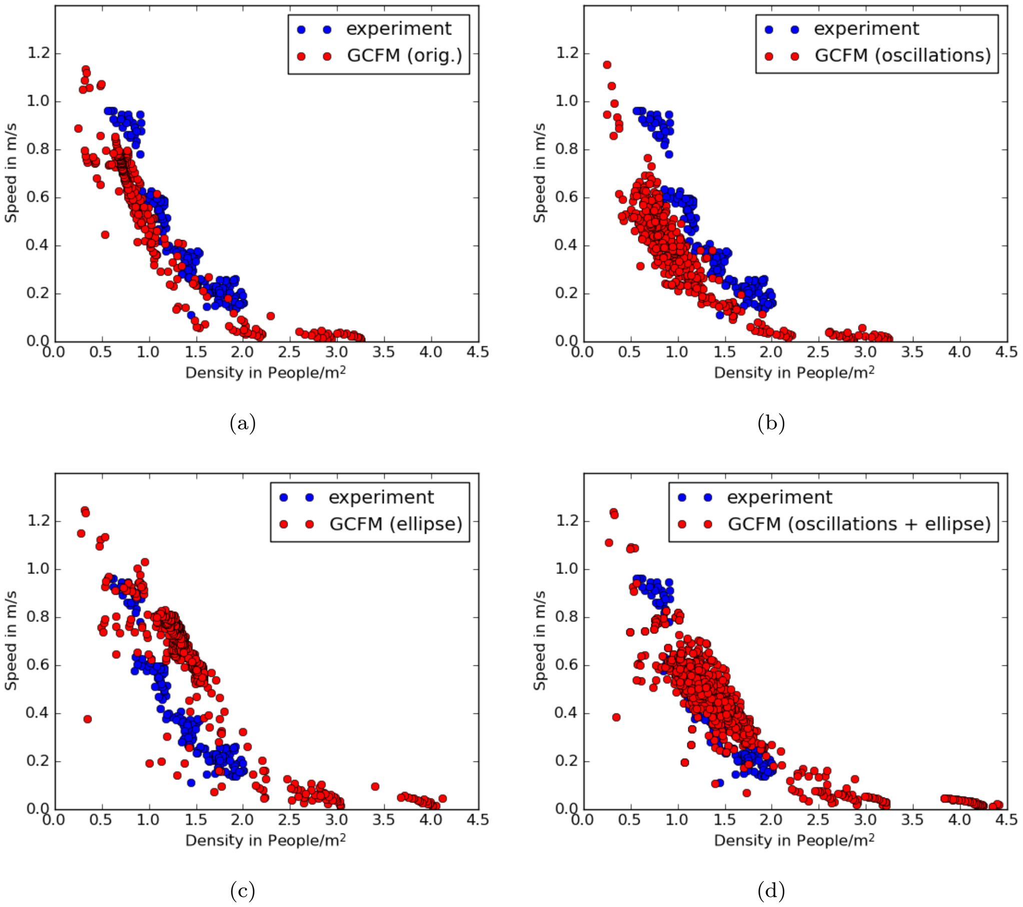
In Figure 5.4, we depict resulting fundamental diagrams in comparison to experimental data [24]. Figure 5.4(a) shows the fundamental diagram for the original GCFM. In Figure 5.4(b), we superimposed the oscillation force (Equation 5.13), whereas in Figure 5.4(c), we adapt the size of the ellipses of pedestrians. Figure 5.4(d) shows the final model as proposed in Section 5.2 with oscillation force and adapted ellipse size.

When superimposing the oscillation force, the speed decreases, since pedestrians require more space for lateral oscillation. In the second case, when we just decrease the size of the pedestrians to a reasonable size, the speed increases due to less space requirements. Finally, when incorporating both modifications, the fundamental diagram fits well to experimental data.

Next, we compare simulated trajectories with real trajectories obtained from the Hermes dataset. For that purpose, we consider 30 seconds of the video (01:00 s - 01:30 s) which corresponds to the time of highest pedestrian density (see Figure 4.8(a)). We use the initial positions and velocities of all pedestrians present at time 01:00 s (62 pedestrians) and simulate their trajectories using the original GCFM as well as the modified GCFM with oscillation force and adapted ellipse size. Figure 5.5 shows real trajectories (green) and simulated trajectories (red).

Here, it becomes obvious that using the original GCFM, pedestrians do not perform lateral swaying. However, when using the proposed modified version of the GCFM, lateral oscillations can be observed. Note that there are differences between real trajectories and simulated trajectories using the modified GCFM, since we do not know the phase of lateral oscillation. Instead, we assume that the phase ϕ_0 in Equation 5.13 is zero.

In order to compare simulated trajectories and real trajectories, we consider the differences in trajectory lengths as well as the Hausdorff distance. Denoting two trajectories of length N and M as $R = (r_1, \ldots, r_N)$ and $S = (s_1, \ldots, s_M)$, respectively, we compute the difference in trajectory lengths as $|N - M|$.



(a)

(b)

(c)

(d)

Figure 5.4: Fundamental diagrams for comparing our proposed model and the original GCFM with regard to experimental data [24]. Pedestrian trajectories are simulated in a corridor (26m × 1m) using the (a) original GCFM, (b) superimposing the oscillation force, (c) adapting the ellipse sizes of pedestrians, and (d) using both modifications.

The Hausdorff distance is computed as [101]:

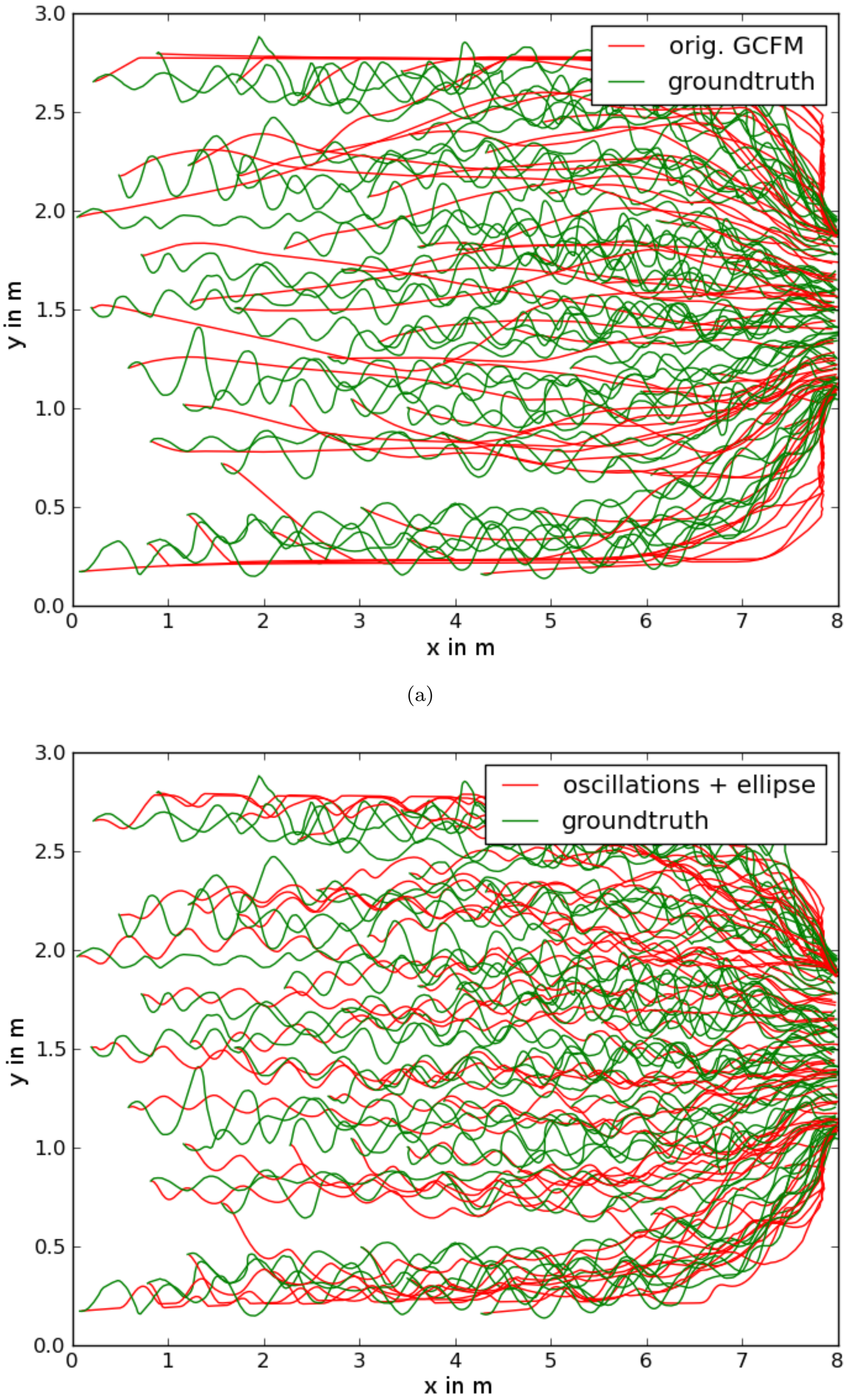
$$H(R,S) = \max(D(R,S), D(S,R)) \tag{5.21}$$

$$D(R,S) = \max_{i=0\dots N}\big(\min_{j=0\dots M} d(r_i, s_j)\big) \tag{5.22}$$

where $d(r_i, s_j)$ denotes the Euclidean distance.

Table 5.2 shows mean, standard deviation and maximum of the Hausdorff distance as well as mean, standard deviation and maximum value of the differences in trajectory lengths. Obviously, the modified GCFM outperforms the original GCFM.

In order to evaluate the proposed model qualitatively, we show that it reproduces self-organizing phenomena that are typically observed in pedestrian dynamics [2, 12, 102]. In particular, we show the effect of lane formation. Therefore, we simulate 300 pedestrians walking in a corridor with opposing walking directions. In the topview shown in Figure



(a)

(b)

FIGURE 5.5: Comparison of simulated trajectories to ground truth trajectories obtained from the Hermes dataset. In green, we depict the ground truth trajectories of 62 pedestrians. For simulating their trajectories, we take their initial positions and velocities and simulate their movements using the (a) original generalized centrifugal force model [27] and (b) our proposed model where we superimpose an oscillation force for including lateral oscillations and adapt the ellipse of pedestrians modeling space requirements. Note that we set the phase of lateral oscillation to zero.

		orig. GCFM	mod. GCFM
Hausdorff	Mean	0.29	0.26
	Standard Deviation	0.12	0.1
	Maximum	0.73	0.64
Length	Mean	120	87
	Standard Deviation	99	61.7
	Maximum	311	233

Table 5.2: Results of comparing real trajectories with simulated trajectories using the original GCFM and our modified version with oscillation force and adapted ellipse sizes. We compute mean, standard deviation and maximum of the Hausdorff distance as well as differences in trajectory lengths.

5.6, we depict pedestrians going to the left and to the right in green and red, respectively. Pedestrian streams of opposing motion direction form lanes in order to minimize interactions with other pedestrians and to minimize avoidance maneuvers. Note that preferences of pedestrians for one side in order to avoid collisions (country-dependent) is not taken into account in the simulation. Hence, lane formation is just a result of repulsive forces between pedestrians.

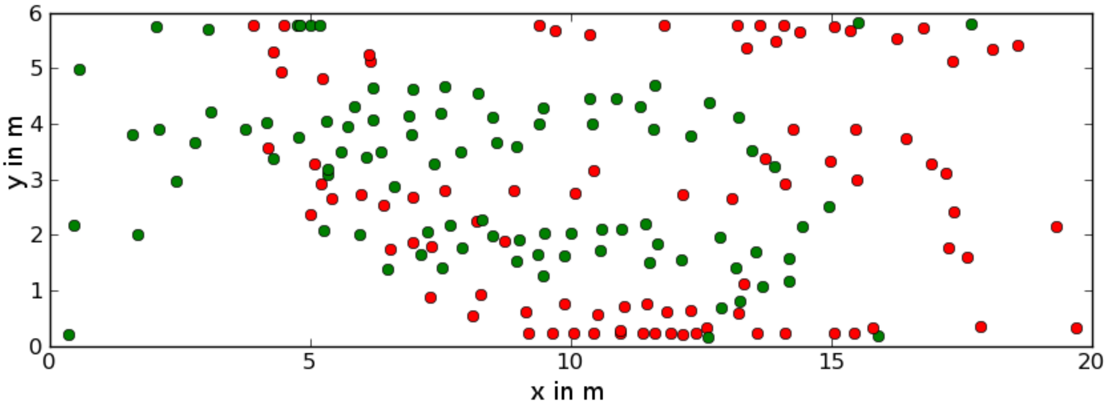


Figure 5.6: Topview of a simulation with 300 pedestrians walking into opposing directions resulting in lane formation. Pedestrians going to the left are depicted in green, whereas pedestrians moving to the right are depicted in red.

5.3.2 Detection of Dangerous Mass Behavior in Simulated Videos

The Loveparade disaster stands out for it has been recorded by seven security cameras over three hours. However, the lack of data is a major problem when developing and evaluating techniques for detecting dangerous mass behavior. Naturally, it is often difficult or even impossible to carry out experiments.

Works towards the combination of artificial life and computer graphics for evaluation of computer vision methods based on generated synthetic data have been reported recently. In [103] and references therein, Qureshi and Terzopoulos propose a framework which integrates simulation and visualization of pedestrian behavior in a virtual environment and combines it with computer vision techniques for object tracking in a virtual camera network. This approach has several advantages: In comparison to real test scenarios, it is a lot easier to simulate different scenarios under different conditions. Computer vision techniques can be tested intensively with varying parameters. Moreover, ground truth data is directly available and privacy issues do not have to be taken into account. Having in mind that computer vision methods also have to be evaluated on real data, the combination of simulation and computer graphics can help to improve and evaluate computer vision algorithms.

In our context, it is obviously difficult or impossible to carry out real experiments. Thus, we employ pedestrian simulation and computer graphics to generate artificial videos which are then analyzed by our methods for detecting unusual events from video. Obviously, it is of great importance that the resulting synthetic videos are as realistic as possible. Hence, we apply the modified generalized centrifugal force model for simulating pedestrian trajectories which has been shown to produce realistic trajectories. For visualizing the simulated pedestrians, we employ 3D animated models of walking humans.

We consider two scenarios: In the first scenario, we simulate 500 pedestrians on a platform of a subway station entering a train. Since only one door of the train is open, the pedestrian density on the platform increases. The second scenario consists of two sub-scenarios taking place at a Christmas market. In one sub-scenario, a critical situation arises due to a fire causing people to escape from the site. In the second sub-scenario, an exit is partially blocked causing high pedestrian densities.

For each scenario, we analyze the simulated pedestrian trajectories and relate measures from pedestrian dynamics to the visual feature computed from the resulting synthesized videos. Next, we apply the sequential change-point detection algorithm to detect critical situations while varying different parameters.

Scenario 1: Subway Station

The first scenario takes place at a subway station where 500 people are walking on the platform and finally enter a train. Figure 5.7 shows a floor plan of the scenario and Figure 5.8 screenshots from two different viewpoints. At the beginning of the simulation, all people are randomly placed in an area of 20 m x 20 m. Then they walk to the platform where a train has stopped. However, just one door (width: 1.20 m) is open forming a bottleneck for the pedestrians and leading to high densities in front of the door.

This scenario is rather similar to the experimental dataset discussed in Section 4.1.2, the Loveparade dataset and scenario 2b. In these scenarios, pedestrians are walking in a corridor which is narrowed by a bottleneck. In contrast, the bottleneck in scenario 1 is located at a side wall of the corridor. Hence, pedestrians are not walking straight ahead to the camera, but turn to the left in order to enter the train.

In order to simulate movements of the pedestrians, we employ the modified generalized centrifugal force model proposed in Section 5.2 and use animated 3D models of walking humans to synthesize videos from two different viewpoints depicted in Figure 5.7. The resulting videos have a resolution of 750 x 500 pixels and are 13:07 min long.

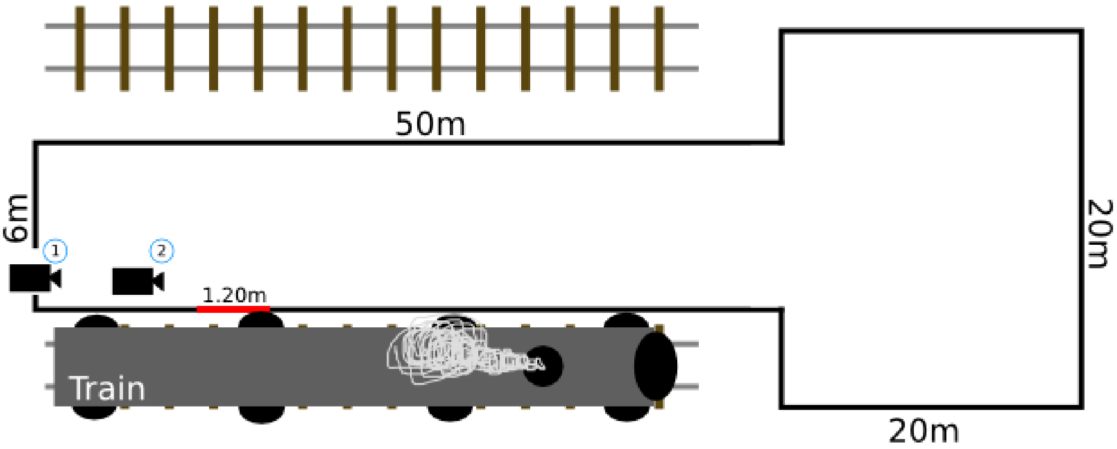
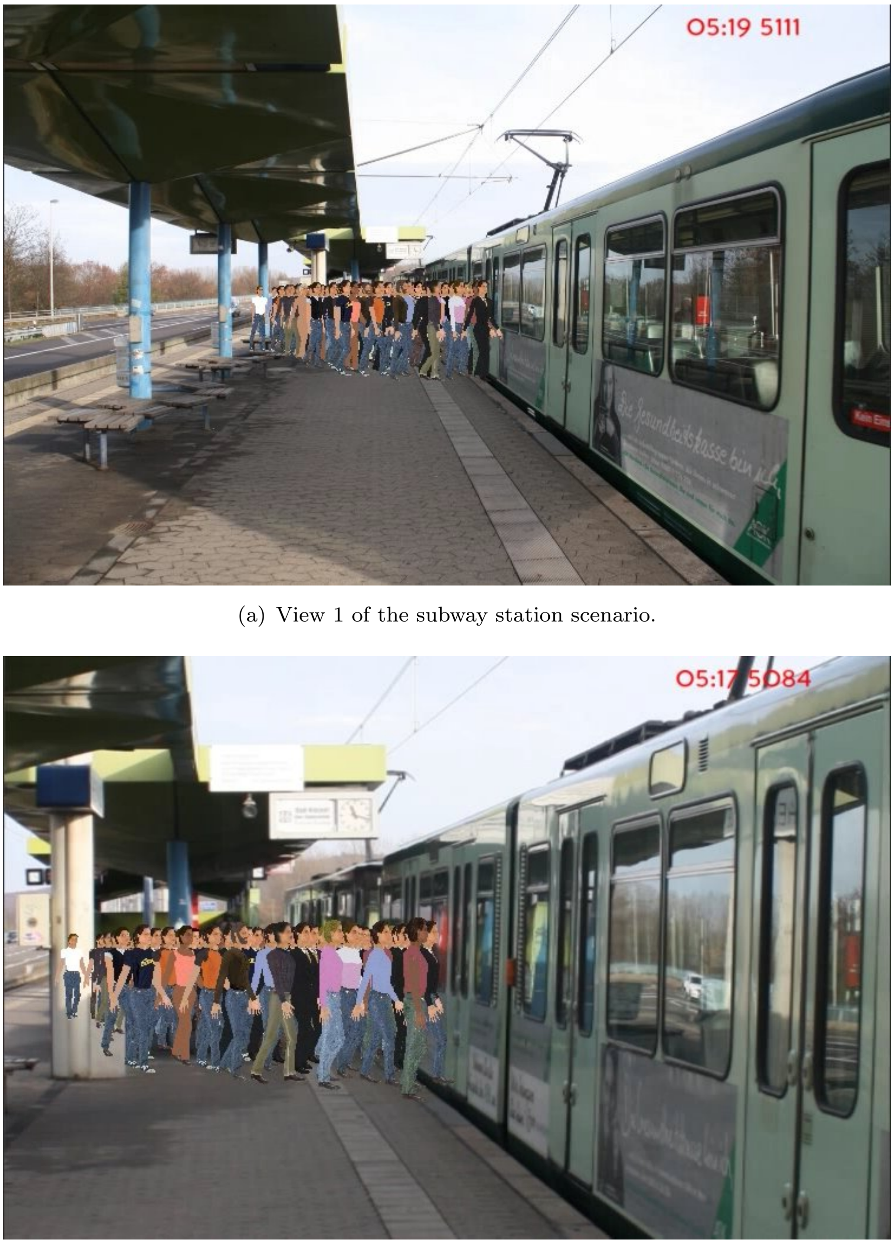


FIGURE 5.7: Floor plan of scenario 1. First, 500 pedestrians are randomly placed in an area of 20 m x 20 m which is depicted on the right in this figure. Then, they walk to the platform and enter the train through the only door which has a width of 1.20 m. As indicated in this sketch, we synthesize videos from two different viewpoints.

We analyze the simulated trajectories in terms of average speed and density over time. Figures 5.9(a) and 5.9(b) show the development of both measures over time where the measurement area is located near the door of the train. At the beginning of the video, the average speed is approximately 1.4 m/s but quickly drops to a low level of 0.4 m/s as the first pedestrians try to pass the bottleneck. Similarly, the pedestrian density near the entrance door of the train is low and increases to a value of 2 pedestrians per square meter. During the next minutes of the video, the average speed and the density remain constant with only a few fluctuations. Then, towards the end of the video, the density decreases leading to increasing speeds.

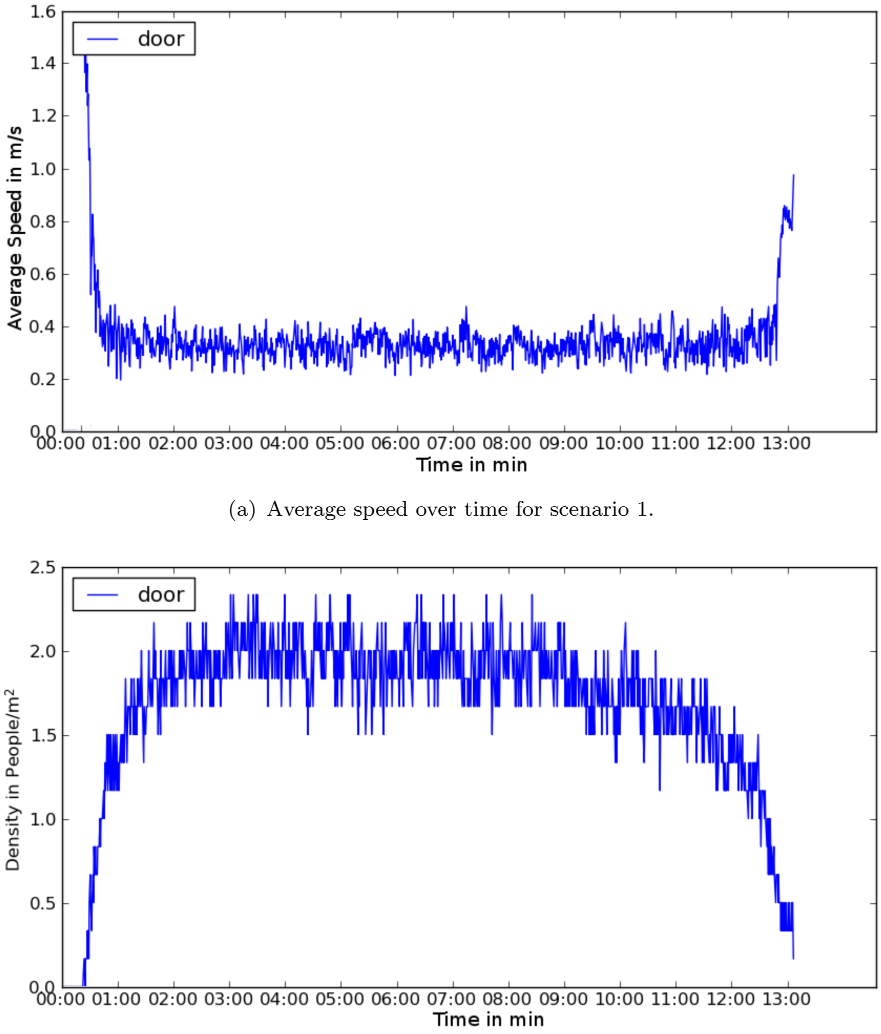
Next, we extract the feature as proposed in Section 3.3.1 and detect change-points by applying our sequential change-point detection method. Figures 5.10 and 5.11 show the development of the feature over time. Comparing these figures to Figure 5.9 reveals that



(a) View 1 of the subway station scenario.

(b) View 2 of the subway station scenario.

Figure 5.8: Screenshots of synthesized videos for two views of scenario 1. In this scenario, 500 people are walking on the platform of a subway station and enter a train. See Figure 5.7 for the floor plan of the scenario and the viewpoints.



(a) Average speed over time for scenario 1.

(b) Density over time for scenario 1.

FIGURE 5.9: Average speed and density over time for scenario 1.

the feature behaves similar to the pedestrian density. In contrast to that, the feature value in scenario 2b, the Loveparade data set and the Hermes dataset decreases for increasing pedestrian densities. In scenario 1, however, the feature value increases for increasing densities, since pedestrians are not walking straight ahead to the camera, but turn to the left in order to enter the train.

In the lower part of Figures 5.10 and 5.11, we show the change-point detection results for different values of l and k, the number of historical data used for estimating a reference distribution and the time span between the current observation and historical data, respectively. In all parameter combinations, the system successfully detects the time when the density reaches its highest level as well as the time when the pedestrian density decreases again. Note the time delay of the alarm when using different parameters for k and l.

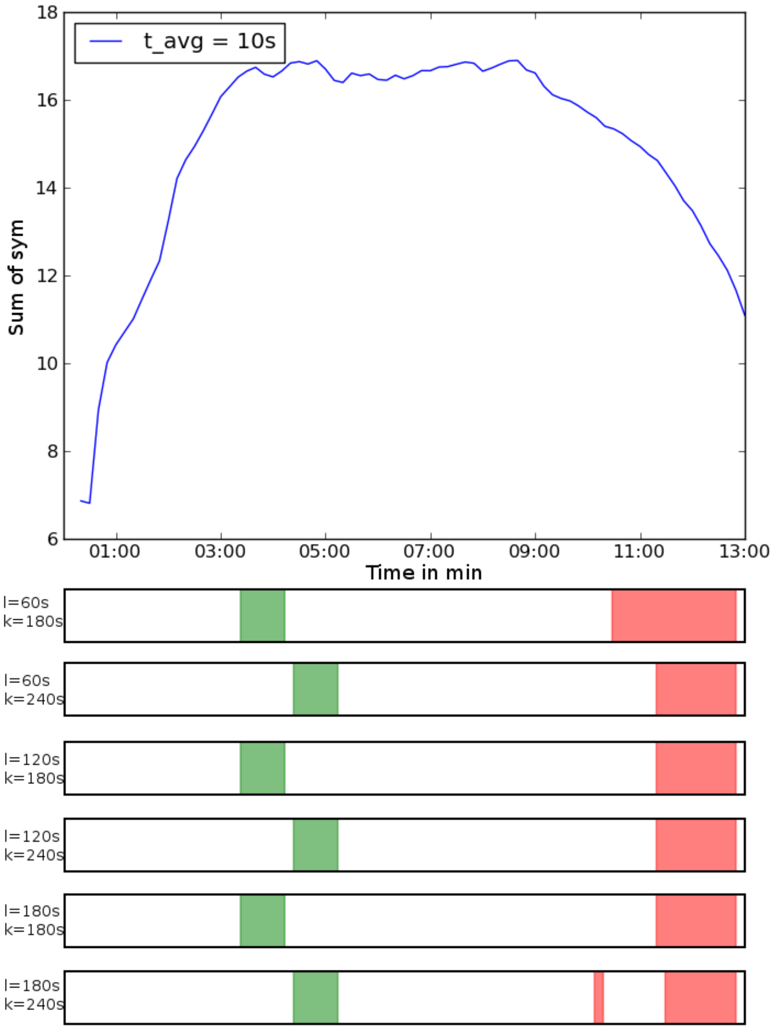


FIGURE 5.10: Detection results on a simulated video of scenario 1 for viewpoint 1 and $t_{avg} = 10s$. The development of the feature measuring the mirror symmetry of the optical flow histograms is depicted in the upper part of the figure, whereas the detection results of the sequential change-point detection algorithm are depicted in the lower part for different parameters k and l. Here, l is the number of historical data used for estimating a reference distribution and suitable thresholds, whereas k is the time span between the current observation and historical data. Red regions indicate decreasing values, whereas green regions indicate increasing values. The system successfully detects the time when the first pedestrians have reached the bottleneck and the time when most of them have passed the bottleneck.

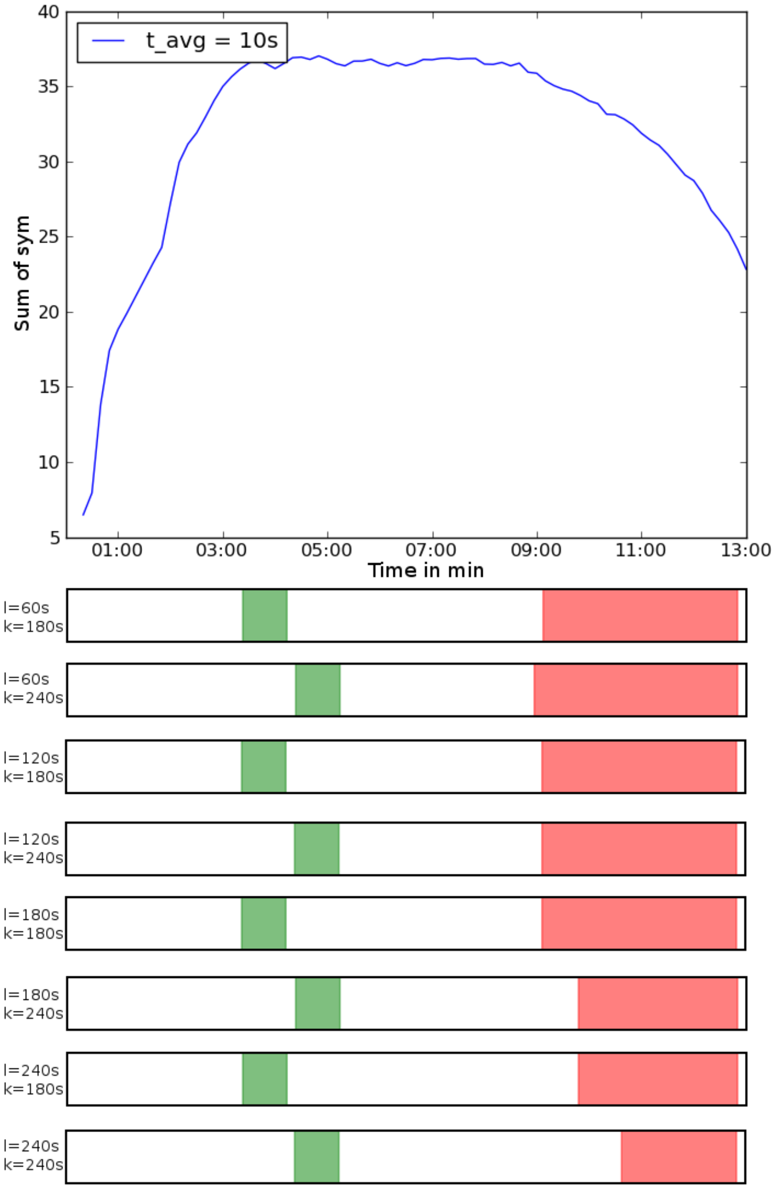


FIGURE 5.11: Detection results on a simulated video of scenario 1 for viewpoint 2 and $t_{avg} = 10s$. The development of the feature measuring the mirror symmetry of the optical flow histograms is depicted in the upper part of the figure, whereas the detection results of the sequential change-point detection algorithm are depicted in the lower part for different parameters k and l. Here, l is the number of historical data used for estimating a reference distribution and suitable thresholds, whereas k is the time span between the current observation and historical data. Red regions indicate decreasing values, whereas green regions indicate increasing values. The system successfully detects the time when the first pedestrians have reached the bottleneck and the time when most of them have passed the bottleneck.

Scenario 2: Christmas Market

The second scenario takes place at a Christmas market and is composed of two subscenarios: a fire at the market and an ambulance car blocking an exit. The floor plan of the Christmas market is depicted in Figure 5.13(a). The corridor is 6m wide and 20m long and has exits on both sides. In both scenarios, 300 people are strolling on the Christmas market. They are walking with a desired speed $v_i^0 = 0.5m/s$ to a randomly chosen exit. An exemplary screenshot of the resulting synthesized video is given in Figure 5.12(a).

In scenario 2a, after 48 seconds, a fire breaks out and blocks exit 1, see Figure 5.12(b) for a screenshot and Figure 5.13(b) for the floor plan. Now, people increase their desired speed v_i^0 to 4 m/s and escape to exit 2.

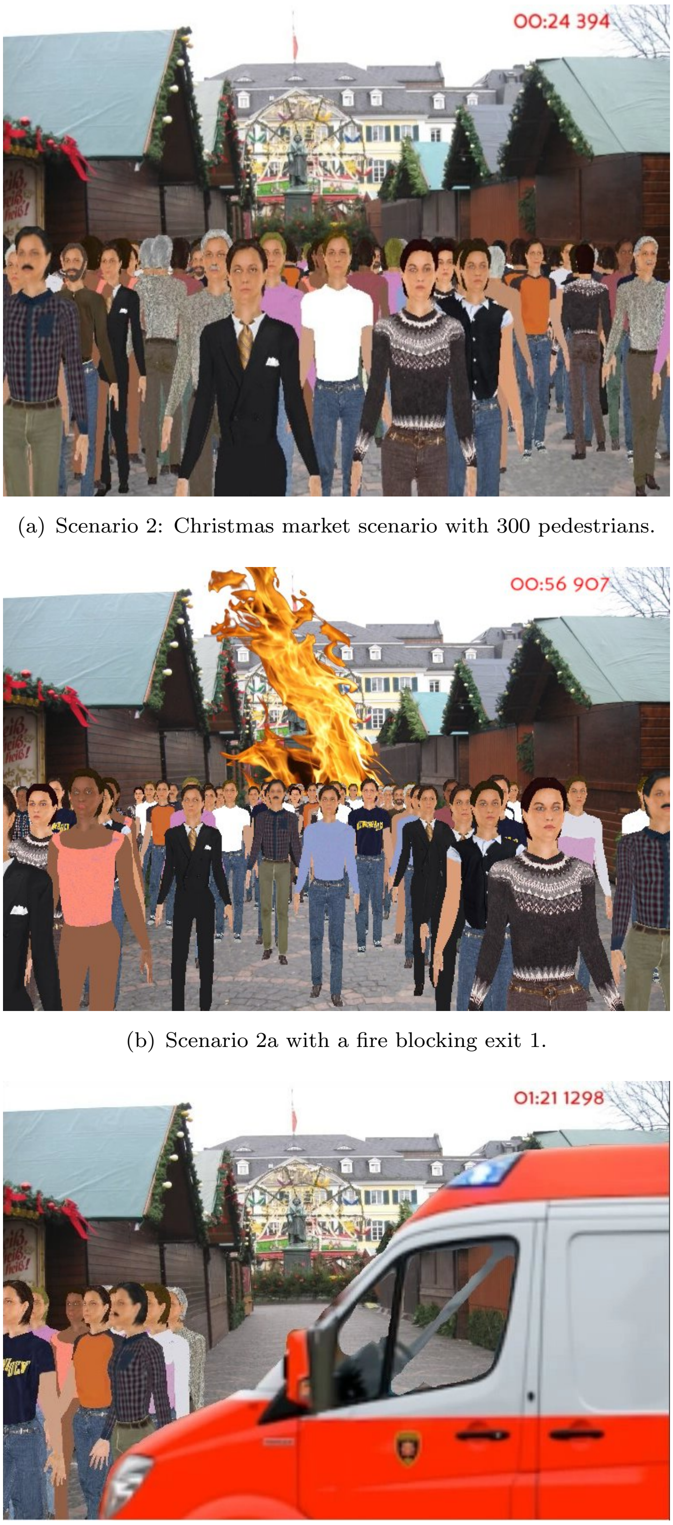
In scenario 2b, an ambulance car partially blocks exit 1 narrowing it to a width of 0.8m (Figures 5.12(c) and 5.13(c)). Here, people keep their desired speed at a constant level $v_i^0 = 0.5$ m/s and continue to walk to their previously chosen exit.

We simulate the trajectories of the pedestrians using the modified generalized centrifugal force model (Section 5.2) and synthesize video sequences using animated 3D models of walking humans. The resulting videos have a frame rate of 16 frames per second, a resolution of 750 x 500 pixels and are 01:06 min (scenario 2a) and 01:42 min (scenario 2b) long.

Scenario 2a: Fire at Christmas Market

By analyzing the simulated trajectories in terms of average speed and density, we obtain Figure 5.14 depicting the average speed and density over time. Both measures are obtained within a measurement area near exit 1 as well as within a second measurement area near exit 2. See Figure 5.13 for the positions of the measurement areas. Figure 5.14(a) shows the average speed in scenario 2a over time. In this scenario, after 48 seconds a fire breaks out causing the people to increase their speed and to escape from the site through exit 2. The change-point is distinctly and visibly in Figure 5.14(a) showing average speeds up to 3.5 m/s, whereas it is not visible in the density over time (Figure 5.14(b)).

Figures 5.15 and 5.16 show the development of the proposed visual feature for different camera viewpoints. See Figure 5.13 for the positions of the camera. They both show similar shapes leading to comparable results when applying the sequential change-point detection method. In the lower part of each figure, we depict the resulting alarms of our system for different parameters of k and l. Here, red regions indicate decreasing feature values, whereas green regions indicate increasing values. For all parameter combinations,

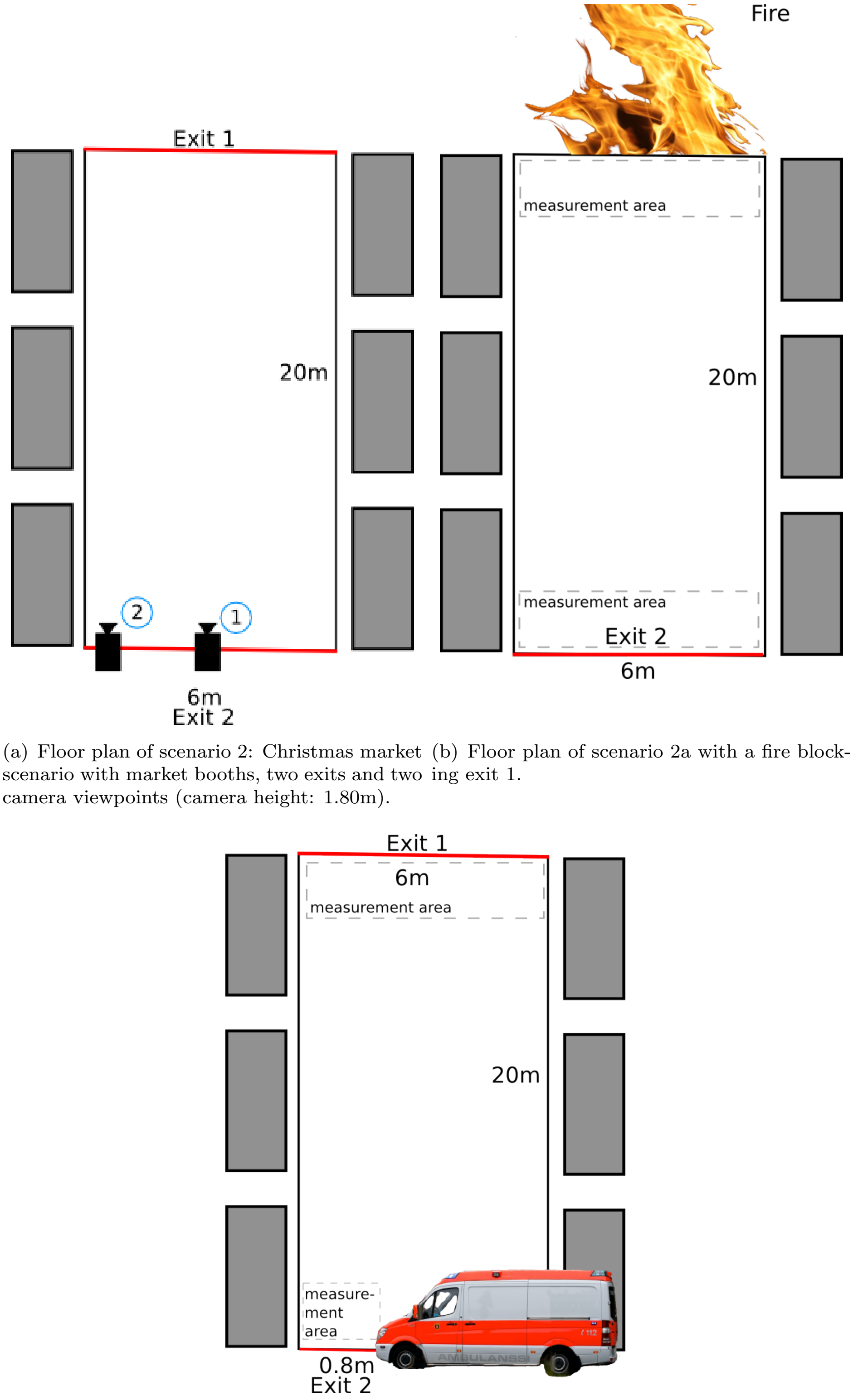


(a) Scenario 2: Christmas market scenario with 300 pedestrians.

(b) Scenario 2a with a fire blocking exit 1.

(c) Scenario 2b with an ambulance car blocking exit 2 partially.

FIGURE 5.12: Screenshots of synthesized videos of scenarios 2a and 2b. In both scenarios, people are strolling on the Christmas market (Figure (a)). They are walking slowly with a velocity of 0.5 m/s. See Figure 5.13 for the floor plan of the scenario. In scenario 2a (Figure (b)), a fire breaks out blocking exit 1. In scenario 2b (Figure (c)), an ambulance blocks exit 2 partially, thus narrowing the exit to a width of 0.8 m. Image credit: Henry Mühlpfordt, Marko Vallius.

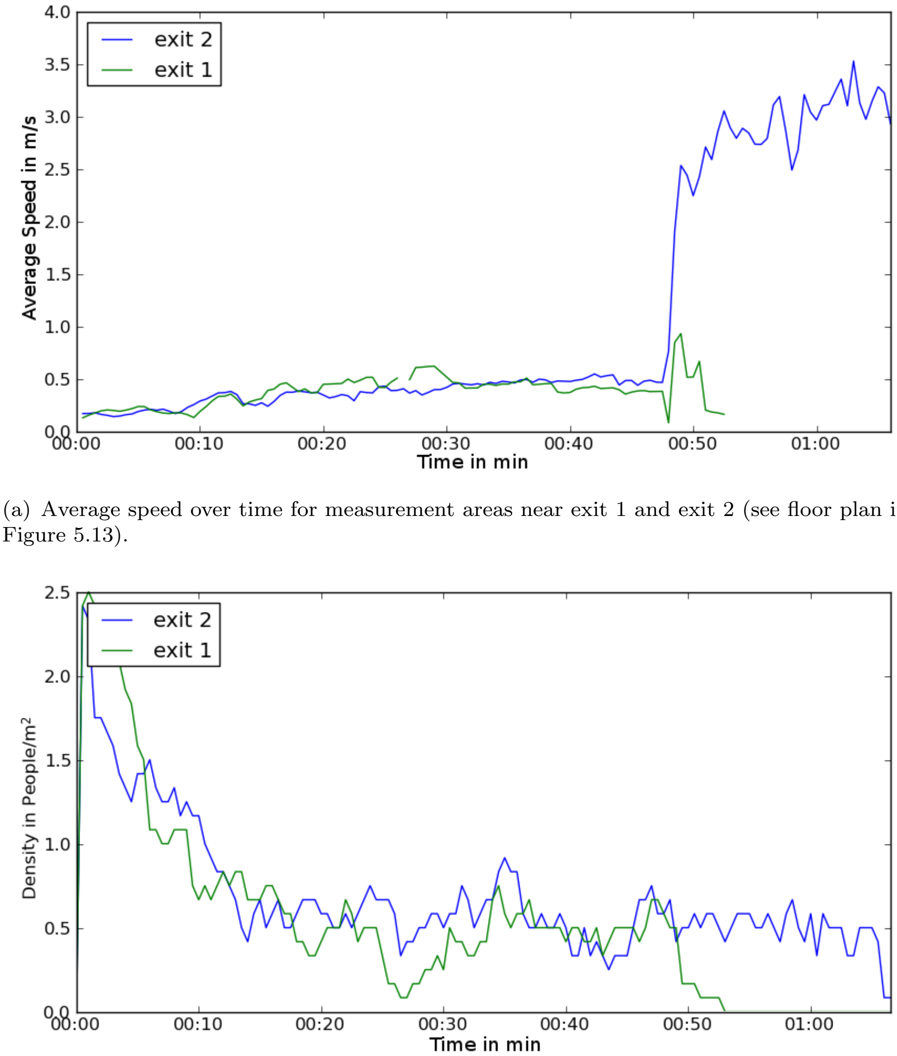


(a) Floor plan of scenario 2: Christmas market scenario with market booths, two exits and two camera viewpoints (camera height: 1.80m).

(b) Floor plan of scenario 2a with a fire blocking exit 1.

(c) Floor plan of scenario 2b with an ambulance car blocking exit 2 partially.

Figure 5.13: Scenario 2. Floor plans of the Christmas market scenarios with Christmas market booths. Image credit: Henry Mühlpfordt, Marko Vallius.



(a) Average speed over time for measurement areas near exit 1 and exit 2 (see floor plan in Figure 5.13).

(b) Density over time for measurement areas near exit 1 and exit 2 (see floor plan in Figure 5.13).

FIGURE 5.14: Average speed and density over time for scenario 2a.

the system successfully detects the time when the fire breaks out. Here, it detects increasing feature values indicating that there is motion into one distinct direction. In addition to that, Figure 5.15 shows that the system also detects increasing feature values at 00:35 min for camera viewpoint 1 which is caused by an increasing pedestrian density near exit 2 visible in Figure 5.14(b). However, often, it also reports a false alarm at the beginning of the video, since the system has not yet integrated enough values. Note that the curves in Figures 5.15 and 5.16 have a high degree of correlation to the average speed in Figure 5.14(a).

Scenario 2b: Exit at Christmas Market Partially Blocked

Similar to scenario 2a, we analyze the simulated trajectories in terms of average speed and density and obtain Figure 5.17 depicting the average speed and density over time.

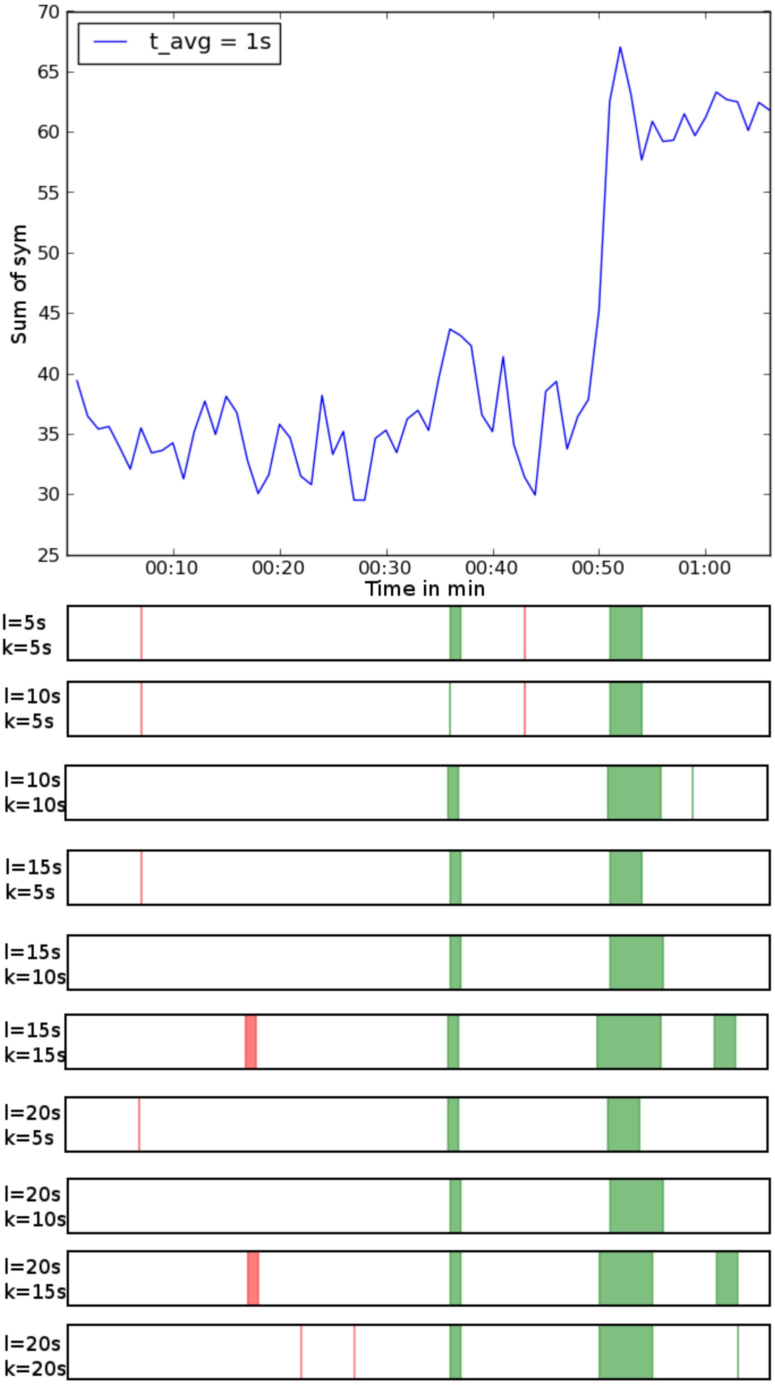


Figure 5.15: Detection results on a simulated video of scenario 2a for viewpoint 1 and $t_{avg} = 1s$. The development of the feature measuring the mirror symmetry of the optical flow histograms is depicted in the upper part of the figure, whereas the detection results of the sequential change-point detection algorithm are depicted in the lower part for different parameters k and l. Here, l is the number of historical data used for estimating a reference distribution and suitable thresholds, whereas k is the time span between the current observation and historical data. Red regions indicate decreasing *sym* values, whereas green regions indicate increasing values. The system successfully detects the time when the fire breaks out at 00:48 min (with a short delay of a few seconds).

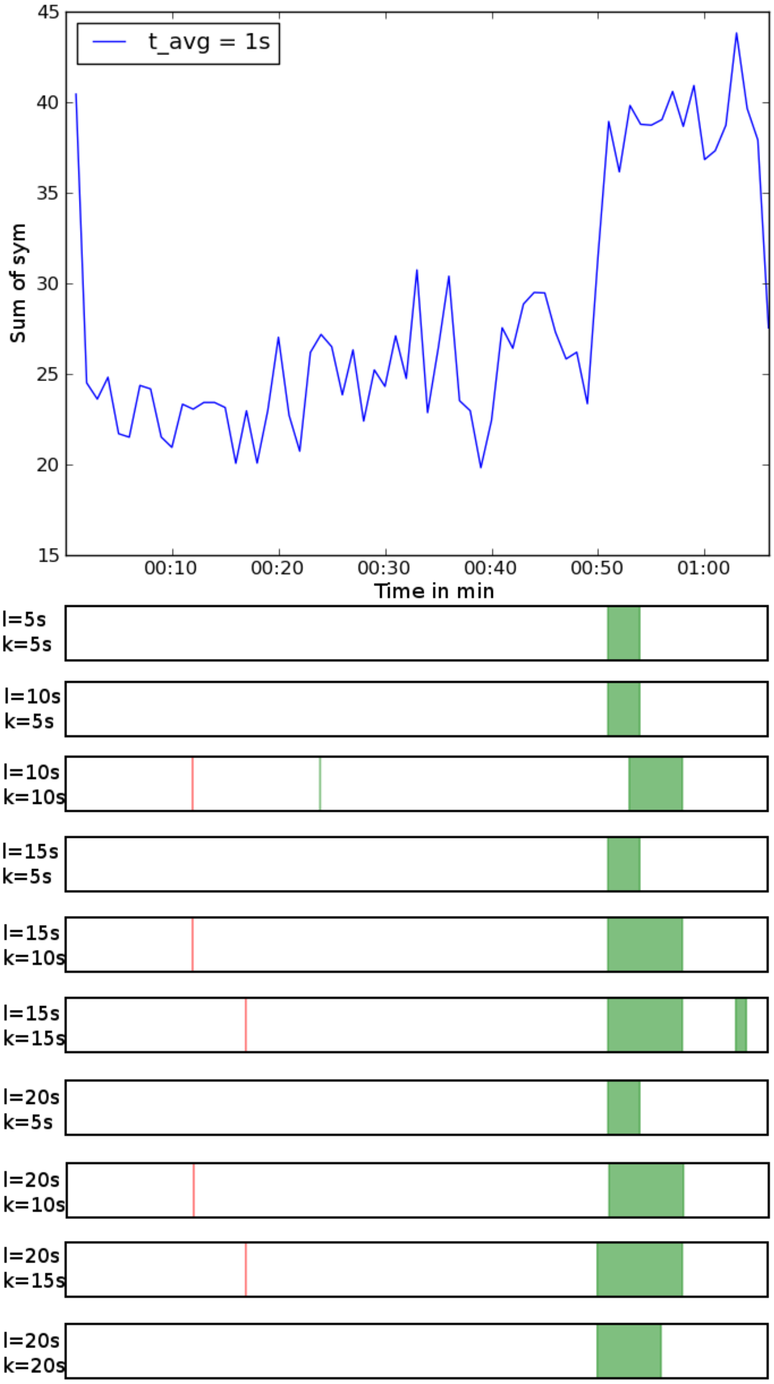
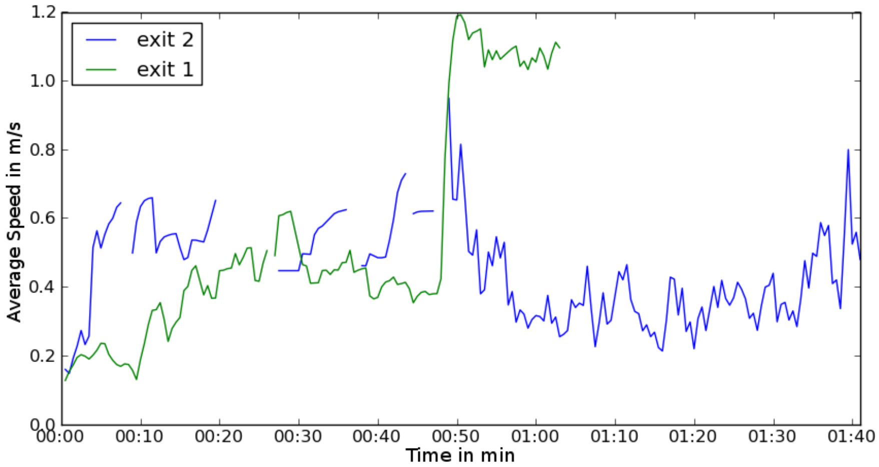
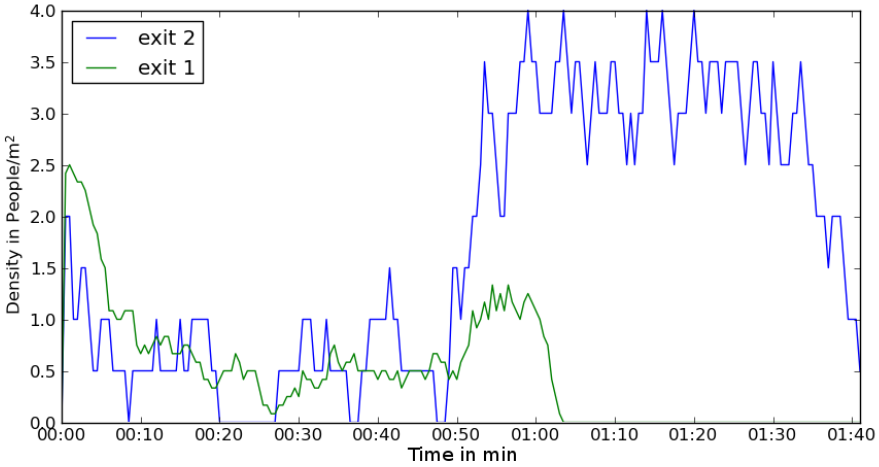


FIGURE 5.16: Detection results on a simulated video of scenario 2a for viewpoint 2 and $t_{avg} = 1s$. The development of the feature measuring the mirror symmetry of the optical flow histograms is depicted in the upper part of the figure, whereas the detection results of the sequential change-point detection algorithm are depicted in the lower part for different parameters k and l. Here, l is the number of historical data used for estimating a reference distribution and suitable thresholds, whereas k is the time span between the current observation and historical data. Red regions indicate decreasing feature values, whereas green regions indicate increasing values. The system successfully detects the time when the fire breaks out at 00:48 min (with a short delay of a few seconds).



(a) Average speed over time for measurement areas near exit 1 and exit 2 (see floor plan in Figure 5.13).



(b) Density over time for measurement areas near exit 1 and exit 2 (see floor plan in Figure 5.13).

Figure 5.17: Average speed and density over time for scenario 2b.

See Figure 5.13 for the positions of the two measurement areas. In scenario 2b, exit 2 is partially blocked after 48 seconds which is clearly visible in Figure 5.17(b). Here, the density reaches values up to 4 people per square meter near exit 2.

In Figures 5.18 and 5.19, we show the development of the feature for the two camera viewpoints (see Figure 5.13) of scenario 2b. Comparing the visual feature with the parameters depicted in Figure 5.17 reveals that it decreases when the pedestrian density reaches a high level indicating a congested area.

In the lower parts of Figures 5.18 and 5.19, the results of the sequential change-point detection algorithm are depicted for different values for k and l. When the ambulance car partially blocks exit 2 after 48 seconds, the system repeatedly reports alarms with decreasing feature values for all parameter combinations. However, for the second viewpoint, these alarms are not as prominent as for the first camera viewpoint. Additionally, the system detects more false alarms as for viewpoint 1.

Summarizing, our system successfully detects the change-points in all synthesized videos caused by congestions (scenario 1 and 2b) and sudden motion changes of pedestrians (scenario 2a). However, at the beginning, the system reports false alarms, as it has not yet integrated enough data.

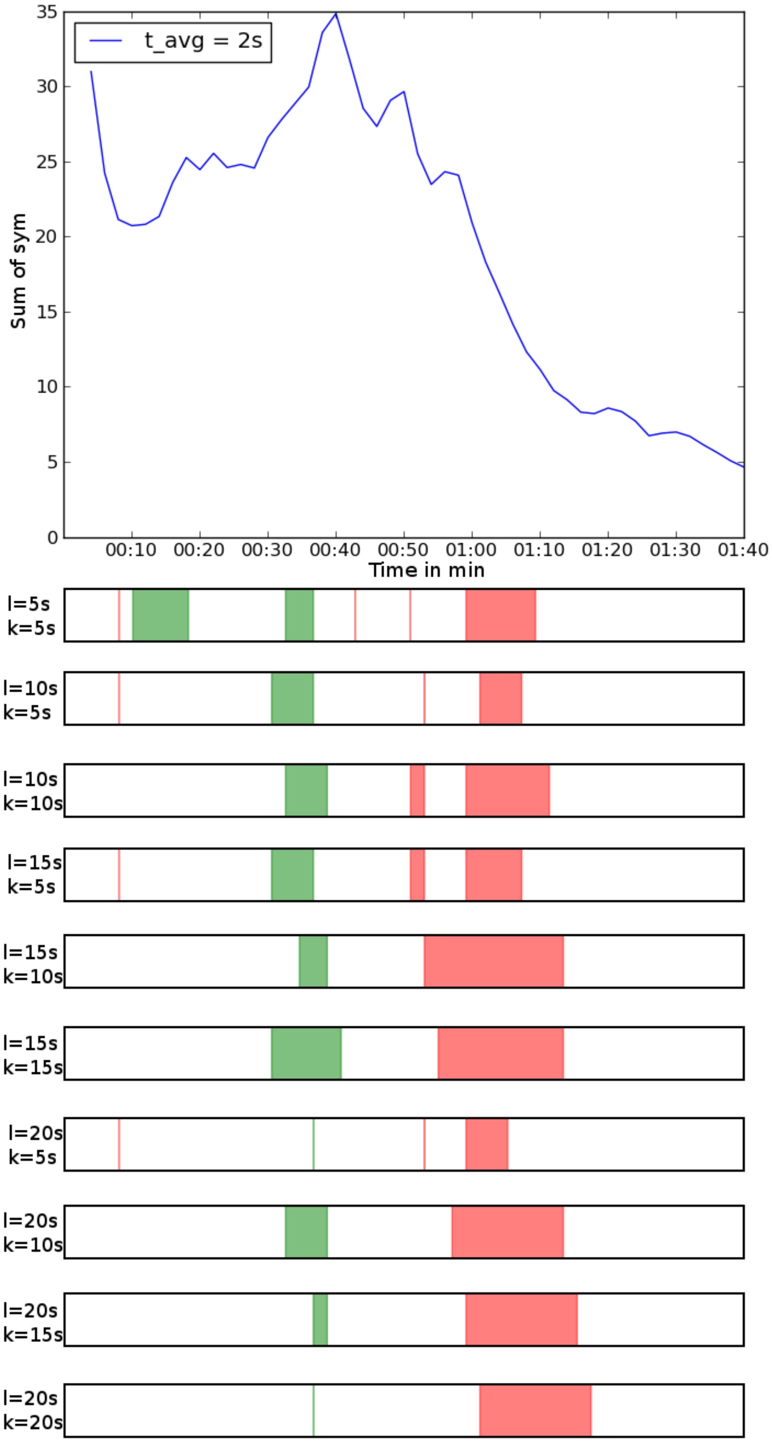


FIGURE 5.18: Detection results on a simulated video of scenario 2b for viewpoint 1 and $t_{avg} = 2s$. The development of the feature measuring the mirror symmetry of the optical flow histograms is depicted in the upper part of the figure, whereas the detection results of the sequential change-point detection algorithm are depicted in the lower part for different parameters k and l. Here, l is the number of historical data used for estimating a reference distribution and suitable thresholds, whereas k is the time span between the current observation and historical data. Red regions indicate decreasing values, whereas green regions indicate increasing values. The system successfully detects the time when the exit is narrowed.

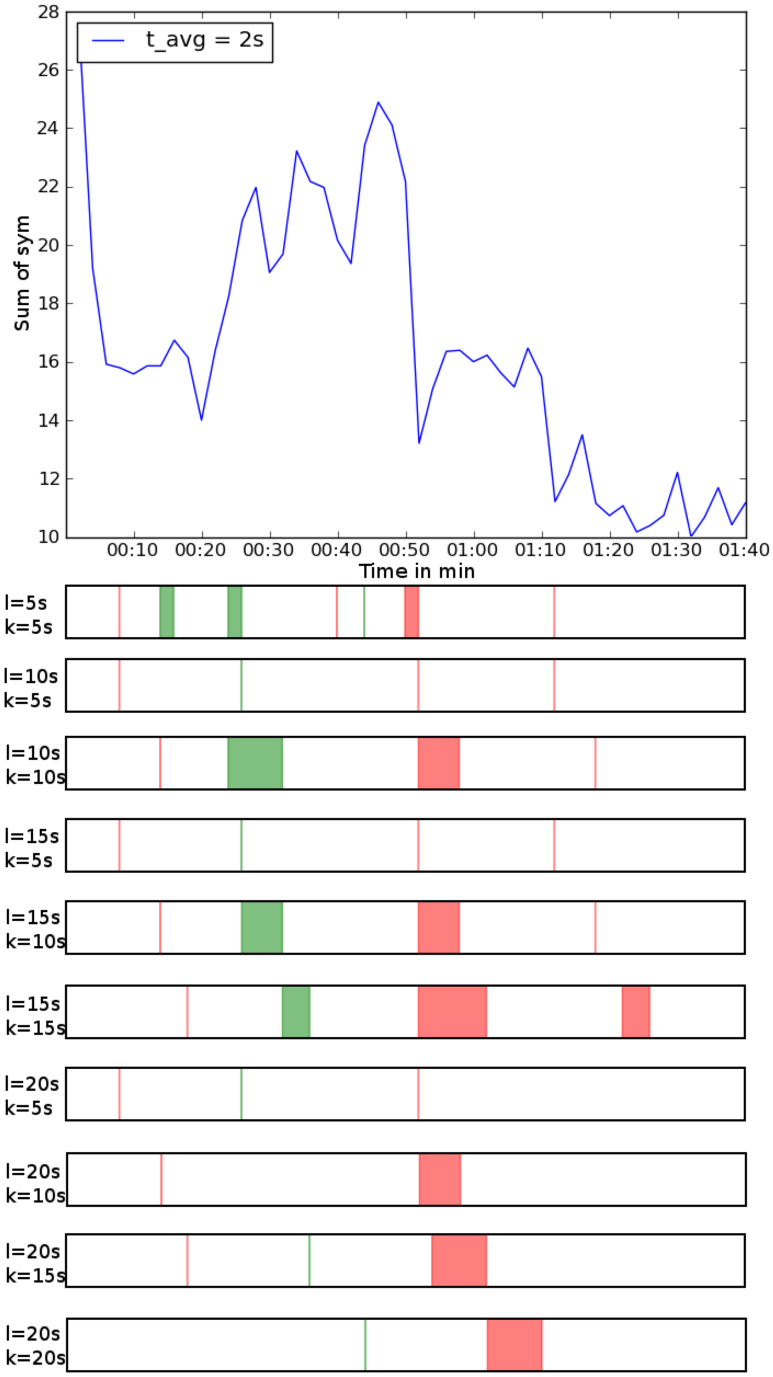


FIGURE 5.19: Detection results on a simulated video of scenario 2b for viewpoint 2 and $t_{avg} = 2s$. The development of the feature measuring the mirror symmetry of the optical flow histograms is depicted in the upper part of the figure, whereas the detection results of the sequential change-point detection algorithm are depicted in the lower part for different parameters k and l. Here, l is the number of historical data used for estimating a reference distribution and suitable thresholds, whereas k is the time span between the current observation and historical data. Red regions indicate decreasing values, whereas green regions indicate increasing values. The system successfully detects the time when the exit is narrowed.

Chapter 6

Conclusion and Outlook

The stampede at the Loveparade 2010 is a recent example of a terrible crowd disaster where, in spite of all precautions, many people died during a mass event. Pedestrian dynamics have been studied intensively for more than four decades. There are two major directions of research in this context. First of all, experimental studies are conducted in order to understand human behavior in crowds and mass evacuation situations which are used to develop physical models of pedestrian dynamics. Second, physical models of crowd behavior are used in simulations that are intended to identify potentially hazardous locations in a building or an event location.

A research direction that is closely related to the aforementioned research domains is computer vision, where researchers develop methods for pedestrian detection and pedestrian tracking in crowded scenes. However, in the past, these research directions did not have any connections. Recently, works of different communities influence each other. For instance, computer vision techniques are employed in order to get insights into motion behavior of pedestrians by tracking people and obtaining measures used in pedestrian dynamics such as pedestrian flow or density. Another example is the integration of pedestrian motion models into computer vision algorithms where these models replace or complement conventional motion models in order to guide pedestrian trackers.

Following this trend of bringing together the aforementioned research directions, the work at hand adopts ideas and techniques from pedestrian dynamics, computer vision, pedestrian simulation and computer graphics. Firstly, we studied pedestrian behavior by analyzing trajectories obtained from an experiment and identified lateral oscillations as a characteristic of human gait in congested areas. This oscillation is a result of the low velocity of people in a congested crowd and is caused by people stepping from one foot to the other in order to keep their balance. Secondly, making use of these observations, we developed an automatic computer vision system to the detection of critical situations in a crowd. Thirdly, we also integrated these observations into models for simulation of pedestrian motion by introducing an oscillation force resulting in more realistic simulations. Finally, using the proposed motion model together with techniques from computer graphics, we were able to simulate crowds threatened by hypothetical dangers and generate artificial videos of this scenarios. These video sequences were then used to test and evaluate our proposed computer vision techniques.

Due to the lack of data showing critical situations in human crowds, there are a few studies of crowd disasters. In fact, this work is the first study of the Loveparade stampede which investigates a real-world incident from a computer vision point of view. Furthermore, to the best of our knowledge, our system is the first video-based system to detect critical situations in crowds in real-time. The extensive video footage of the Loveparade stampede enabled us to evaluate the system on a real-world incident. The proposed system successfully detects unusual events such as crossing of cars and a congestion in particular. In addition to that, we have evaluated our system on real-world videos from a subway station as well as on videos of experiments conducted under laboratory conditions and synthetic videos that we generated using our proposed pedestrian simulation model.

Our approach is based on histograms of flow vector magnitude and direction and computes symmetry measures from histograms of optical flow vectors. The system learns typical feature values and alarms security personnel whenever it detects deviations from the learned values. Our method is applicable in many different situations and is mostly independent of local conditions and camera viewpoints. Moreover, since the proposed approach is based on dense optical flow fields and simple measures derived therefrom, it works in real-time and thus meets a crucial prerequisite in video surveillance. Finally, since our method does not necessitate identification or tracking of people, it preserves the privacy of the pedestrians being monitored.

Currently, our system detects unusual events and critical situations and alerts security personnel to evaluate the situation and to take necessary actions. However, the decision if and how to react is solely based on the expert knowledge of the security personnel. Here, a system for decision support could help to prevent accidents. In particular, short-term simulations could be conducted in order to simulate different options and to choose optimal actions. For instance, a simulation module could evaluate how pedestrian streams should be redirected in order to improve the situation or which evacuation strategy should be selected. For that purpose, the simulation has to integrate dynamic information such as pedestrian densities in different regions which could be obtained by computer vision techniques. Beyond that, simulations could even be used in order to foresee critical situations and to prevent them. In both scenarios, two fundamental conditions have to be met: First, the coupling of a simulation module and an analysis module providing input for the simulation has to be established. Second, both the analysis of the current situation as well as the simulations have to be conducted in real-time.

These considerations illustrate the need of collaboration, convergence and exchange of different research areas. However, the application domains range from all kinds of gatherings and large buildings to public spaces and public transportation facilities. Despite of all challenges, the application of modern technologies in these scenarios will be of high value, in particular, when considering growing populations and increasing popularity of mass gatherings.

Bibliography

[1] Lopavent. Dokumentation Loveparade. `www.dokumentation-loveparade.com`. last visited July 1, 2011.

[2] D. Helbing and A. Johansson. Pedestrian, crowd and evacuation dynamics. In *Encyclopedia of Complexity and Systems Science*. 2009.

[3] K. Still. Crowd disasters. `http://crowdmodelling.com/CrowdDisasters.html`. last visited January 12, 2012.

[4] J. Fruin. The Causes and Prevention of Crowd Disasters. In *Engineering for Crowd Safety*, 1993.

[5] E. Wallace, C. Diffley, and J. Aldridge. Good practice for the management and operation of town centre CCTV. In *Conf. on Security and Detection*, 1997.

[6] CARETAKER Consortium. Caretaker puts knowledge to good use. *Mobility, The European Public Transport Magazine*, 18(13):14–18, 2008.

[7] D. Chowdhury, L. Santen, and A. Schadschneider. Statistical physics of vehicular traffic and some related systems. *Physics Reports*, 329(4–6):199–329, 2000.

[8] K. Still. *Crowd Dynamics.* PhD thesis, University of Warwick, 2000.

[9] D. Helbing, A. Johansson, and H. Al-Abideen. Dynamics of crowd disasters: An empirical study. *Physical Review E*, 75(4):046109–1–7, 2007.

[10] J. Jacques, S. Musse, and C. Jung. Crowd analysis using computer vision techniques. *Signal Processing Magazine*, 27:66–77, 2010.

[11] B. Zhan, D. Monekosso, P. Remagnino, S. Velastin, and L. Xu. Crowd analysis: a survey. *Machine Visions and Applications*, 19(5–6):345–357, 2008.

[12] D. Helbing and P. Molnár. Social force model for pedestrian dynamics. *Physical Review E*, 51(5):4282–4286, 1995.

[13] C. Burstedde, K. Klauck, A. Schadschneider, and J. Zittartz. Simulation of pedestrian dynamics using a 2-dimensional cellular automaton. *Physica A*, 295(3–4): 507–525, 2001.

[14] G. G. Løvås. Modeling and simulation of pedestrian traffic flow. *Transportation Research B*, 28(6):429–443, 1994.

[15] R. L. Hughes. The flow of human crowds. *Annual Review of Fluid Mechanics*, 35 (35):169–182, 2003.

[16] R. L. Hughes. A continuum theory for the flow of pedestrians. *Transportation Research B*, 36(6):507–535, 2002.

[17] A. Treuille, S. Cooper, and Z. Popovic. Continuum crowds. *ACM Trans. on Graphics*, 25(3):1160–1168, 2006.

[18] A. Hanisch, J. Tolujew, K. Richter, and T. Schulze. Online simulation of pedestrian flow in public buildings. In *Simulation Conf.*, 2003.

[19] J. Tolujew and F. Alcalá. A mesoscopic approach to modeling and simulation of pedestrian traffic flows. In *Europ. Simulation Multiconf.*, 2004.

[20] T. Kretz, A. Grünebohm, M. Kaufman, F. Mazur, and M. Schreckenberg. Experimental study of pedestrian counterflow in a corridor. *Journal of Statistical Mechanics*, (10):P10001, 2006.

[21] D. Helbing, I. Farkas, and T. Vicsek. Simulating dynamical features of escape panic. *Nature*, 407(6803):487–490, 2000.

[22] D. R. Parisi, M. Gilman, and H. Moldovan. A modification of the social force model can reproduce experimental data of pedestrian flows in normal conditions. *Physica A*, 388(17):3600–3608, 2009.

[23] T. Lakoba, D. J. Kaup, and N. M. Finkelstein. Modifications of the Helbing-Molnár-Farkas-Vicsek social force model for pedestrian evolution. *Simulation*, 81 (5):339–352, 2005.

[24] A. Seyfried, B. Steffen, and T. Lippert. The fundamental diagram of pedestrian movement revisited. *Journal of Statistical Mechanics*, P10002:339–352, .

[25] M. Höcker, P. Milbradt, and A. Seyfried. Simulations of pedestrian dynamics and model adjustments: A reality-based approach. In *Int. Conf. Traffic and Granular Flow*, 2009.

[26] W. J. Yu, R. Chen, L. Y. Dong, and S. Q. Dai. Centrifugal force model for pedestrian dynamics. *Phsical Review E*, 72(2):026112, 2005.

[27] M. Chraibi and A. Seyfried. Generalized centrifugal-force model for pedestrian dynamics. *Phsical Review E*, 82(4):046111, 2010.

[28] W. Yu and A. Johansson. Modeling crowd turbulence by many-particle simulations. *Phsical Review E*, 76(4):046105, 2007.

[29] U. Weidmann. Transporttechnik der Fußgänger. Schriftenreihe des IVT 90, ETH Zürich, 1992.

[30] X. Liu, W. Song, and J. Zhang. Extraction and quantitative analysis of microscopic evacuation characteristics based on digital image processing. *Physica A*, 388(13): 2717–2726, 2009.

[31] A. Johansson, D. Helbing, H. Al-Abideen, and S. Al-Bosta. From crowd dynamics to crowd safety: A video-based analysis. *Advances in Complex Systems*, 11(04): 497–527, 2008.

[32] M. Boltes, A. Seyfried, B. Steffen, and A. Schadschneider. Automatic extraction of pedestrian trajectories from video recordings. In *Pedestrian and Evacuation Dynamics 2008*.

[33] U. Chattaraj, A. Seyfried, and P. Chakroborty. Comparison of Pedestrian Fundamental Diagram Across Cultures. *Advances in Complex Systems*, 12(03):393–405, 2009.

[34] J. F. Morrall, L. L. Ratnayake, and P. N. Seneviratne. Comparison of CBD Pedestrian Characteristics in Canada and Sri Lanka. *Transportation Research Record*, 1294:57–61, 1991.

[35] T. Fujiyama and N. Tyler. An explicit study on walking speeds of pedestrians on stairs. In *Int. Conf. Mobility and Transport for Elderly and Disabled People*, 2004.

[36] T. Kretz, A. Grünebohm, and M. Schreckenberg. Experimental study of pedestrian flow through a bottleneck. *Journal of Statistical Mechanics*, page P10014, 2006.

[37] S. P. Hoogendoorn and W. Daamen. Pedestrian behavior at bottlenecks. *Transportation Science*, 39:147–159, 2005.

[38] A. Seyfried and A. Schadschneider. Empirical results for pedestrian dynamics at bottlenecks. In *Int. Conf. on Parallel Processing and Applied Mathematics*, 2009.

[39] W. Daamen and S. Hoogendoorn. Capacity of doors during evacuation conditions. *Procedia Engineering*, 3(0):53 – 66, 2010.

[40] A. Seyfried, M. Boltes, J. Kähler, W. Klingsch, A. Portz, T. Rupprecht, A. Schadschneider, B. Steffen, and A. Winkens. Enhanced empirical data for the fundamental diagram and the flow through bottlenecks. In *Pedestrian and Evacuation Dynamics 2008.* .

[41] A. Seyfried, A. Portz, and A. Schadschneider. Phase coexistence in congested states of pedestrian dynamics. In *Int. Conf. on Cellular automata for research and industry*, 2010.

[42] B. Ashe and T. J. Shields. Analysis and Modelling of the Unannounced Evacuation of a Large Retail Store. *Fire and Materials*, 23:333–336, 1999.

[43] H. Weckman, S. Lehtimäki, and S. Männikö. Evacuation of a Theatre: Exercise vs Calculations. *Fire and Materials*, 23:357–361, 1999.

[44] H. Klüpfel, T. Meyer-König, and M. Schreckenberg. Comparison of an Evacuation Exercise in a Primary School to Simulation Results. In *Traffic and Granular Flow '01*, 2002.

[45] W. Hu, T. Tan, L. Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *Transactions on Systems, Man and Cybernetics.*, 34 (3):334–352, 2004.

[46] T. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126, 2006.

[47] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Computing Surveys*, 38(4):13, 2006.

[48] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *ECCV*, 2004.

[49] B. Leibe, B. Seeman, and B. Schiele. Pedestrian detection in crowded scenes. In *CVPR*, 2005.

[50] A. C. Davies, J. H. Yin, and S. A. Velastin. Crowd monitoring using image processing. *Electronics and Communication Engineering Journal*, 7:37–47, 1995.

[51] B. P. Lo and S. A. Velastin. Automatic congestion detection system for underground platforms. In *Intelligent Multimedia, Video and Speech Processing*, 2001.

[52] A. Chan, Z. Liang, and N. Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *CVPR*, 2008.

[53] A. N. Marana, L. F. Costa, R. A. Lotufo, and S. A. Velastin. On the efficacy of texture analysis for crowd monitoring. In *Symposium on Computer Graphics, Image Processing, and Vision*, 1998.

[54] B. A. Boghossian and S. A. Velastin. Motion-based machine vision techniques for the management of large crowds. In *Electronics, Circuits and Systems*, 1999.

[55] X. Wu, G. Liang, K. Lee, and Y. Xu. Crowd density estimation using texture analysis and learning. In *Robotics and Biomimetics*, 2006.

[56] J. M. Gryn, R. P. Wildes, and J. K. Tsotsos. Detecting motion patterns via direction maps with application to surveillance. *Computer Vision and Image Understanding*, 113(2):291–307, 2009.

[57] B. Krausz and C. Bauckhage. Action recognition in videos using nonnegative tensor factorization. In *ICPR*, 2010.

[58] X. Wang, X. Ma, and E. Grimson. Unsupervised activity perception by hierarchical bayesian models. In *CVPR*, 2007.

[59] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz. Robust real-time unusual event detection using multiple fixed-location monitors. *Transactions on Pattern Analysis and Machine Intelligence*, 30(3), 2008.

[60] R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behavior detection using social force model. In *CVPR*, 2009.

[61] Y. Yang, J. Liu, and M. Shah. Video scene understanding with multi-scale analysis. In *ICCV*, 2009.

[62] I. Saleemi, L. Hartung, and M. Shah. Scene understanding by statistical modeling of motion patterns. In *CVPR*, 2010.

[63] J. Kim and K. Grauman. Observe locally, infer globally: a space-time MRF for detecting abnormal activities with incremental updates. In *CVPR*, 2009.

[64] L. Kratz and K. Nishino. Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In *CVPR*, 2009.

[65] M. Breitenstein, H. Grabner, and L. van Gool. Hunting Nessie – real-time abnormality detection from webcams. In *Int. Workshop on Visual Surveillance*, 2009.

[66] E. Andrade, S. Blunsden, and R. Fisher. Modelling crowd scenes for event detection. In *ICPR*, 2006.

[67] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos. Anomaly detection in crowded scenes. In *CVPR*, 2010.

[68] S. Ali and M. Shah. A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis. In *CVPR*, 2007.

[69] S. Ali and M. Shah. Floor fields for tracking in high density crowd scenes. In *ECCV*, 2008.

[70] P. Widhalm and N. Brändle. Learning major pedestrian flows in crowded scenes. In *ICPR*, 2010.

[71] S. Wu, B. Moore, and M. Shah. Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes. In *CVPR*, 2010.

[72] J. Kerridge, S. Keller, T. Chamberlain, and N. Sumpter. Collecting pedestrian trajectory data in real-time. In *Pedestrian and Evacuation Dynamics*, 2005.

[73] K. H. Lee, M. G. Choi, Q. Hong, and J. Lee. Group behavior from video: a data-driven approach to crowd simulation. In *SIGGRAPH/Eurographics Symposium on Computer animation*, 2007.

[74] G. Antonini, S. V. Martinez, M. Bierlaire, and J. P. Thiran. Behavioral priors for detection and tracking of pedestrians in video sequences. *Int. Journal of Computer Vision*, 69(2):159–180, 2006.

[75] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *ICCV*, 2009.

[76] S. Pellegrini, A. Ess, M. Tanaskovic, and L. van Gool. Wrong turn - no dead end: a stochastic pedestrian motion model. In *Int. Workshop on Socially Intelligent Surveillance and Monitoring*, 2010.

[77] E. Andrade and R. Fisher. Simulation of crowd problems for computer vision. In *Int. Workshop on Crowd Simulation*, 2005.

[78] P. Allain, T. Corpetti, and N. Courty. Crowd flow characterization with optimal control theory. In *ACCV*, 2009.

[79] N. Courty and T. Corpetti. Crowd motion capture. *Computer Animation and Virtual Worlds*, 18:361–370, 2007.

[80] S. R. Musse, C. R. Jung, J. Jacques, and A. Braun. Using computer vision to simulate the motion of virtual agents. *Computer Animation and Virtual Worlds*, 18:83–93, 2007.

[81] A. Schadschneider, W. Klingsch, H. Klüpfel, T. Kretz, C. Rogsch, and A. Seyfried. Evacuation dynamics: Empirical results, modeling and applications. In *Encyclopedia of Complexity and Systems Science*. 2009.

[82] V. M. Predtechenskii and A. I. Milinskii. Planning for foot traffic flow in buildings. *Amerind Publishing*, 1969.

[83] M. Mori and H. Tsukaguchi. A new method for evaluation of level of service in pedestrian facilities. *Transportation Research A*, 21(3):223–234, 1987.

[84] B. Steffen and A. Seyfried. Methods for measuring pedestrian density, flow, speed and direction with minimal scatter. *Physica A*, 389(9):1902–1910, 2010.

[85] A. Portz and A. Seyfried. Modeling stop-and-go waves in pedestrian dynamics. In *Parallel Processing and Applied Mathematics*, 2010.

[86] A. Portz and A. Seyfried. Analyzing stop-and-go waves by experiment and modeling. In *Pedestrian and Evacuation Dynamics*, 2010.

[87] DFG Research Project. `http://www2.fz-juelich.de/jsc/appliedmath/ped/projects/dfg`. last visited January 12, 2012.

[88] D. Helbing, A. Johansson, and H. Al-Abideen. Crowd turbulence: the physics of crowd disasters. In *Int. Conf. on Nonlinear Mechanics*, 2007.

[89] Spiegel.de. Katastrophe bei der Love Parade. `http://www.spiegel.de/panorama/0,1518,708399,00.html`. last visited January 12, 2012.

[90] S. Utler. Die wundersame Schrumpfung der 1,4 Millionen. `http://www.spiegel.de/panorama/0,1518,709372-7,00.html`. last visited January 12, 2012.

[91] G. Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Scandinavian Conf. on Image Analysis*, 2003.

[92] D. Lee and H. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755), 1999.

[93] C. Thurau and V. Hlavac. Pose primitive based human action recognition in videos or still images. In *CVPR*, 2008.

[94] V. M. De Oca, D. Jeske, Q. Zhang, C. Rendon, and M. Marvasti. A cusum change-point detection algorithm for non-stationary sequences with application to data network surveillance. *Journal of Systems and Software*, 83:1288–1297, 2010.

[95] E. S. Page. Continuous inspection schemes. *Biometrika*, 41:100–115, 1954.

[96] Hermes. BMBF Research Programme. `http://www2.fz-juelich.de/jsc/appliedmath/ped/projects/hermes`. last visited January 12, 2012.

[97] D. Oberhagemann. Personenstromanalysen Loveparade 2010. Technical report, Vereinigung zur Förderung des Deutschen Brandschutzes e.V., 2010.

[98] EVA. BMBF Research Programme. `http://www.vfdb.de/EVA.117.0.html`. last visited January 12, 2012.

[99] O. Junker, V. Strauss, R. Majer, and N. Link. Real-time video analysis of pedestrians to support agent simulation of people behaviour. In *Pedestrian and Evacuation Dynamics*, 2010.

[100] M. Chraibi, A. Seyfried, A. Schadschneider, and W. Mackens. Quantitative description of pedestrian dynamics with a force-based model. In *Int. Conf. on Web Intelligence and Intelligent Agent Technologies*, 2009.

[101] I. N. Junejo, O. Javed, and M. Shah. Multi feature path modeling for video surveillance. In *ICPR*, 2004.

[102] D. Helbing. Collective phenomena and states in traffic and self-driven many-particle systems. *Computational Materials Science*, 30(1-2):180 – 187, 2004.

[103] F. Qureshi and D. Terzopoulos. Smart camera networks in virtual reality. *Proceedings of the IEEE Special Issue on "Distributed Smart Cameras"*, 96(10):1640–1656, 2008.