2014-ENST-0004

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité « Signal et Images »

présentée et soutenue publiquement par

Hajer FRADI

28 January 2014

Nouvelles Méthodes pour l'Étude de la Densité des Foules en Vidéo Surveillance

Directeur de thèse : **Jean-Luc DUGELAY**

Jury

M. Frédéric DUFAUX, Directeur de Recherches, TELECOM ParisTech, France	Président de Jury
M. Carlo REGAZZONI, Professeur, Université de Gène, Italie	Rapporteur
M. François BREMOND, Directeur de Recherches, INRIA Sophia Antipolis, France	Rapporteur
M. Christian FEDORCZAK, Directeur des programmes européens, THALES, France	Examinateur
M. Jean-Luc DUGELAY, Professeur, EURECOM, France	Directeur de Thèse

TELECOM ParisTech
école de l'Institut Télécom - membre de ParisTech

THÈSE

New insights into Crowd Density Analysis in Video Surveillance Systems

Dedicated to my Ph.D. study from 2010.06 to 2014.01

The research presented in this thesis was supported by the European project VIDEOSENSE.

Abstract

Along with the widespread growth of surveillance cameras, computer vision algorithms have played a fundamental role in analyzing the large amount of videos. However, most of the current approaches in automatic video surveillance assume that the observed scene is not crowded, and is composed of easily perceptible components. These approaches are hard to be extended to more challenging videos of highly crowded scenes such as in religious festivals, marathons, sport events, public demonstrations, subways etc., where detecting and tracking individuals is a very difficult task. Therefore, a number of studies have recently begun to focus on the analysis of high-density scenes.

Crowd analysis has recently emerged as an increasingly important and dedicated problem for crowd monitoring and management in the visual surveillance community. Specifically, the estimation of crowd density is receiving a lot of attention and it is of significant interest for crowd safety in order to prevent potentially dangerous situations. In this thesis, our first objective is to address the problems of crowd density estimation (such as people counting, crowd level estimation and crowd motion segmentation) and the second objective is to investigate the usefulness of such estimation as additional information to other video surveillance applications.

Towards the first goal, we focus on the problems related to the estimation and the characterization of the crowd density using low level features in order to avert typical problems in detection of high density crowd, such as dynamic occlusions and clutter. We demonstrate in this dissertation, that the proposed approaches perform better than the baseline methods, either for counting the number of people in crowds, or alternatively for estimating the crowd level. Afterwards, we propose a novel approach for crowd density measure, in which local information at the pixel level substitutes the overall crowd level or person count. Our approach is based on modeling time-varying dynamics of the crowd density using sparse feature tracks as observations of a probabilistic density function.

The second goal of this study is to explore an emerging and promising field of research in crowd analysis which consists of using crowd density as additional information to complement other tasks related to video surveillance in crowded scenes. First, since the application of conventional detection and tracking methods in crowds is of limited success, we use the proposed crowd density measure which conveys rich information about the local distributions of persons in the scene to improve human detection and tracking in videos of high density crowds. Second, we investigate the concept of crowd context-aware privacy protection by adjusting the obfuscation level according to the crowd density. Finally, we employ additional information about the local density together with regular motion patterns as crowd attributes for high level applications such as crowd change detection and event recognition.

Résumé

Avec la propagation des caméras de vidéosurveillance, la vision par ordinateur joue un rôle primordial dans l'analyse des vidéos. Cependant, la majorité des approches actuelles en vidéosurveillance supposent que les scènes sont simples et constituées d'éléments facilement discernables. En effet, ces approches ne sont pas toujours appropriées pour des caméras de surveillance filmant des scènes denses comme les fêtes religieuses, les événements sportifs, les expositions de grande envergure, les passages souterrains destinés à la circulation pédestre, etc. dans lesquelles la détection et le suivi des individus sont une tâche extrêmement difficile. Par conséquent, ces dernières années, on constate qu'il y a davantage de recherche pour analyser des scènes complexes.

Désormais, l'analyse des scènes denses s'impose incontestablement comme une tâche importante pour pouvoir contrôler et gérer les foules. Notamment, on accorde à l'estimation de la densité de la foule et à la sécurité de celle-ci une importance particulière, pour anticiper les débordements potentiellement dangereux. Notre recherche a pour objectifs, d'abord d'apporter des solutions à l'estimation de la densité de la foule (comme le comptage d'individus, l'estimation du niveau de la foule et la segmentation des mouvements de celle-ci). Ensuite, elle vise à prouver l'utilité de cette estimation comme préalable pour d'autres applications de vidéosurveillance.

Concernant le premier objectif, afin de cerner des difficultés relatives à la détection de personnes dans une foule dense, comme c'est le cas lors de l'afflux d'un grand nombre de personnes en même temps, notre recherche se focalise sur l'estimation et la caractérisation de la densité de la foule basée sur un niveau d'analyse bas. Dans un premier temps, nous démontrons que nos diverses approches sont plus adéquates que les méthodes de l'état de l'art que ce soit pour compter le nombre d'individus constituant une foule ou pour estimer le niveau de la foule. Dans un second temps, pour mesurer la densité de la foule, nous proposons une approche innovante dans laquelle une estimation locale au niveau des pixels remplace l'estimation au niveau global de la foule ou le comptage des personnes. Notre approche est basée sur l'utilisation des suivis de caractéristiques visuelles dans une fonction de densité.

Notre recherche a également pour objectif d'explorer un nouveau domaine d'étude qui s'avère prometteur et fructueux pour l'analyse de la foule. Il consiste à utiliser la densité de la foule comme information supplémentaire pour affiner d'autres tâches liées à la vidéosurveillance des scènes denses. D'abord, partant du constat que les méthodes conventionnelles de détection et de suivi de personnes ne sont pas parfaitement adaptées pour l'analyse de la foule, nous avons utilisé la mesure de la densité de la foule qui comporte une description pertinente relative à la répartition spatiale des individus afin d'améliorer leur détection et leur suivi dans des scènes denses. Ensuite, en prenant en compte la notion de la protection de la vie privée dans le contexte de surveillance d'une foule, nous ajustons

le niveau de floutage en fonction de la densité de la foule. Enfin, nous nous appuyons sur l'estimation locale de la densité ainsi que sur des mouvements en tant qu'attributs pour les applications de haut niveau telles que la détection des évolutions dans la foule et la reconnaissance des événements.

Acknowledgements

This thesis would not have been possible without the help, guidance and support of many people, who I would like to acknowledge here.

I am indebted to my thesis advisor Prof. Jean-Luc Dugelay for giving me the opportunity for a PhD. at EURECOM / Telecom ParisTech. Throughout my PhD he provided helpful ideas and encouraging support. He created a vastly positive and enthusiastic working atmosphere that fuelled self-motivation and ambition.

I would like to thank my committee members, the reviewers Prof. Carlo Regazzoni and Dr. François Bremond, and furthermore the examiners Dr. Frédéric Dufaux and Dr. Christian Fedorczak for their precious time, shared positive insight and guidance.

I would like also to show my sincerest gratitude to all my professors at ENSI (Tunisia) for providing me expertise on every subject. Their enthusiasm and dedication to their students were truly inspiring. Mainly, many thanks to Prof. Faouzi Ghorbel for giving me copious amounts of insightful guidance and constant encouragement.

I would like to express my deepest appreciation for Communication Systems Group leaded by Prof. Thomas Sikora in Technische Universitat Berlin for the cordial reception during my visit there. Mainly, I would like to thank Volker Eiselein, and Tobias Senst who shared their brilliance and creativity with me. It was a pleasure to work and exchange with them. My warmest thanks to my colleagues who supported me during my Ph.D. Precisely, I would like to thank Simon Bozonnet, Carmelo Velardo, Rui Min, Xuran Zhao, and Andrea Melle. Also, I thank all those working at EURECOM, they made my stay at EURECOM very pleasant.

I owe my deepest gratitude to my parents, Ali Fradi and Najet Frad, my sister Haifa and my brother Anis for their unwavering encouragement, devotion and love. Lastly, special thanks to my friends for their unwavering friendship, moral and infinite support.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

1.1 Context and Motivation

Crowd denotes a large group of individuals who have gathered closely together. The phenomenon of crowd and its dynamics have been studied in different research disciplines such as sociology, civil, and physics. Nowadays, it becomes one of the most active-oriented research area and attractive topic in computer vision.

The steady population growth with the worldwide urbanization render the crowd phenomenon more frequent. According to UN [1] estimates (see Figure 1.1), the population growth could reach 9.3 billion in 2050. Also, another recent study shows that more than half of the world population are living in densely populated areas.

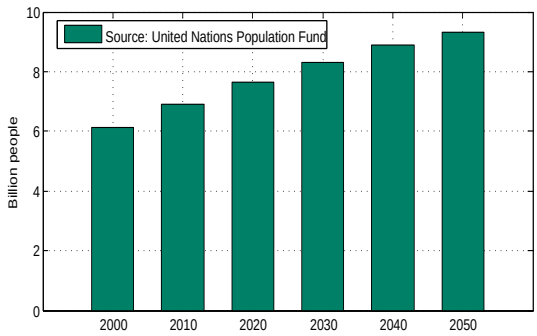


Figure 1.1: *World Population Growth 1950-2050*

Furthermore, studying crowd phenomenon is of great interest mainly with the increasing number of popular events that gather many people such as markets, subways, religious festivals, public demonstrations, parades, concerts, football matches, races, sport events, and high density moving objects like traffic.

In this context, crowd analysis is emerged as major topic for crowd monitoring and management in visual surveillance community. In particular, the estimation of crowd density is receiving much attention and significant interest for safety control. It could be used for developing crowd management strategies to insure public safety by measuring the comfort level in public spaces. Also, its automatic monitoring is extremely important to prevent disasters by detecting potential risk and preventing overcrowd mainly when the number of

[1] `http://www.unfpa.org/public/`

persons flooding some areas exceeds a certain level of crowd (e.g. in some religious and sport events). Many stadium tragedies could illustrate this problem, as well as the Love Parade stampede in Germany and the Water Festival stampede in Colombia. To prevent such deadly accidents, early detection of unusual situations in large scale crowd is required and appropriate decisions for safety control have to be taken to insure assistance and emergency contingency plan. In addition, density estimation of passengers is relevant to economic applications such as optimizing the schedule of public transportation systems and organizing the working hours of employees in shopping malls.

Hence, many recent works in the field of automatic video surveillance have been proposed to address the problem of crowd density analysis. Typically, given a video sequence the objective is to estimate the number of people, or alternatively to estimate the crowd level. These two categories (people counting and crowd level estimation) have been studied separately in the literature, whereas, it exists obvious overlaps due to the fact that the density is defined as the number of persons per unit area. Practically, since techniques based on person detection have some difficulties to operate on videos containing high density of crowds, more sophisticated methods to retrieve density information have been employed. Specifically, recent works mostly bypass the task of people detecting and instead focus on learning a mapping between a set of low level features and the number of persons or the level of the crowd.

During this PhD thesis work, our study is focused on crowd density estimation and its application to others video surveillance applications. In particular, we intend:

- to investigate today's state-of-the-art: where we are in crowd density analysis?
- to address the problems of people counting and density level estimation in crowded scenes, which are of primary interest in surveillance systems.
- to propose a feature vector for crowd density estimation which is robust enough to perform well in different levels of the crowd.
- to improve the accuracy of crowd density estimation results compared to the state-of-the-art.
- to perform comparisons with various features; that enables better investigation of which features are discriminative to the crowd density.
- to extend crowd density estimation to local level by building crowd density maps.
- to prove that in high crowded situations, when the baseline algorithms for person detection and tracking do not perform well, the estimation of crowd density map could significantly improve person detectors and tracks.
- to investigate the usefulness of applying crowd density in privacy context.

- to demonstrate the relevance of using local crowd density which captures the distributions of persons in the scene together with motion information for crowd change detection and event recognition.

A detailed analysis of these problems as well as our contributions in this field are presented in Section 1.2.

1.2 Thesis Contributions

In this thesis, we focus on the problems related to crowd density analysis. In particular, two crucial components have been studied in the literature, which are people counting and crowd level estimation. For the first component, we introduce a novel method, where only a frame-wise normalized feature is used in the regression step. For this purpose, two different approaches have been proposed: The first approach is based on measurements of interest points (this work was published at EUSIPCO 2012), where a perspective normalization and a crowd measure-informed density estimation are introduced into a single feature with the number of moving SIFT points. Then, the correspondence between this feature and the number of persons is learned by Gaussian Process regression. Our approach has been experimentally validated showing more accurate results compared to other regression-based methods. In the second proposed approach (this approach was published at WIFS 2012), we adopt an integration of Gaussian Mixture Model (GMM) background subtraction with an uniform motion model into a single overall system, which has the potential to better segment foreground entities (this integration was published at ICIEV 2012). Therefore, we intend to harness the advantage of incorporating motion model into GMM to obtain high accurate foreground segmentation. Then, the counting is based on measurements of foreground pixels; we propose to apply a perspective map normalization in order to compensate the variations in distance. Also, we apply a crowd measure where FAST local features are synthesized for a global corner density. In this work, we also get better results than other regression-based methods. In addition, we demonstrate the benefits of integrating GMM with motion cue.

After studying people counting problem, where we test some statistical features (number of SIFT interest points, foreground area, and corner density), we address the second component of crowd density which is crowd level classification. For this purpose, we first process image patches in order to generate images with regions of interest. Then, in the feature extraction block, we investigate the descriptive power of Local Binary Pattern (LBP) features for crowd density estimation compared to other texture features. Also, we explore the impact of applying dimensionality reduction techniques in the feature space due to the high dimension of block-based LBP (this work was published at ICME workshops 2013).

Crowd density estimation is a multi-class problem in which the goal is to assign different levels of crowd (the crowd is quantized into 5 levels: free, restricted, dense, very

dense and jammed flow) to local image patches. Since Support Vector Machine (SVM) is designed for binary classification, usually several binary SVMs have to be performed. To maintain low computing complexity for multi-classification problem, we propose an alternative algorithm that involves less binary SVM classifiers, based on reassessing each binary SVM using relevance scores (this work was published at ICIP 2013).
The results show that effective dimensionality reduction (DR) techniques on LBP feature vectors significantly enhance the classification performance compared to high dimensional raw features. Also, by means of comparisons with other texture features, our proposed feature (LBP+DR) has been experimentally validated showing more accurate results with a significant margin. Furthermore, the comparison of our proposed multiclass SVM with two baseline methods highlights the usefulness of our proposed algorithm in terms of accuracy while maintaining less computational cost.

After studying the two aforementioned crowd density forms (i.e. people counting and the level of the crowd density), we propose a novel approach for crowd density measure, in which local information at pixel level substitutes an overall crowd level or a number of people per-frame. Although the forms of people counting and crowd level classification are commonly used in the field of crowd analysis for security reasons, they have the limitation of giving a global information of the entire image and discarding local information about the crowd. We therefore resort to crowd information at local level by computing crowd density maps using local features as an observation of a probabilistic crowd function. The proposed approach also involves a feature tracking step which enables excluding feature points on the background. This process is favorable for the later density function estimation since the influence of features irrelevant to the underlying crowd density is removed. We evaluate our proposed approach on videos from different datasets, and the results demonstrate the effectiveness of using feature tracks for crowd estimation. Furthermore, we include a comparative study between different local features to investigate their discriminative power to the crowd (this work was published at MMSP 2013).

After that, we explore a promising research direction which consists of using crowd density measurement to complement different other applications in video surveillance, such as improving human detection and tracking in crowded scenes, detecting and recognizing crowd events, and boosting the compliance between surveillance and privacy protection in crowds by formulating contextualized privacy protection filters. First, we propose to use the crowd density maps to enhance people detection and tracking algorithms in high crowded scenes where delineating individuals is considered as a challenging task because of spatial overlaps. The idea is based on introducing additional information about crowds and integrating it into the state-of-the-art detector. Our proposed approach applies a scene-adaptive dynamic parametrization using the crowd density measure. It also includes a self-adaptive learning of the human aspect ratio and the perceived height in order to reduce false positive detections. The advantages of incorporating crowd density and adding geometrical constraints to the detection process have been experimentally validated showing better results

(this work was published at AVSS 2013).
Obviously, achieving reliable detection can deeply affect many other applications. To illustrate that, we extend our proposed improved detection algorithm to tracking using the Probability Hypothesis Density (PHD) tracker. The results show an improvement compared to the baseline method which is expected as trackers rely on improved detections (this extension is submitted to Signal Processing Journal: Image Communication).

Second, we propose to use the crowd density maps to adjust the level of privacy protection according to the local needs. In particular, we build adaptive privacy protection filters, in which the privacy level gradually decreases with the crowd density. The idea is based on the observation: the less people are present around a site, the more perceivable and identifiable is a single individual. At the same time, for safety control, video operators need clear visual information in overcrowded areas, where potentially dangerous events could occur. It is therefore reasonable in many applications to reduce the privacy level in crowded areas compared to spaces with isolated individuals (this work was published at DSP 2013).
To demonstrate the effectiveness of these contextualized protection filters, we propose an objective evaluation of privacy and intelligibility trade-off. By leveraging state-of-the-art video surveillance analysis algorithms, such as people counting and matching, we show that our contextualized privacy filters retain good performances on common intelligibility tasks such as people counting and detection. At the same time, such privacy filters are able to significantly reduce the performances of person matching algorithm based on local features, which can potentially expose identity information of the subject being monitored, therefore threatening its privacy (this work was published at ISM 2013).

The last application consists of using a crowd density measure for higher level analysis such as crowd change detection, crowd behavior recognition, and crowd event characterization. While most of the existing works in this field rely on regular motion patterns such as speed and flow direction, we consider that local crowd density is an important cue for early detection and recognition of crowd events and it could complement crowd dynamics (motion) information. Our proposed approach is based on capturing local distributions of persons in the scene together with motion information using feature tracking in order to determine the ongoing crowd events. The experimental results demonstrate the effectiveness of our proposed approach for early detection of crowd change and accurate results for event recognition (this work is accepted at ICPR 2014 and its extension is submitted to Signal, Image and Video Processing Journal, Special issue on *Semantic representations for social behavior analysis in video surveillance systems*).

1.3 Thesis Outline

The work presented within this thesis fits the context of crowd density analysis and video surveillance applications awareness crowd density.

Chapter 2 is dedicated to recall some useful definitions and paradigms related to video surveillance systems with more focus on crowd analysis problem. We start by introducing video surveillance functionalities, then, we present some main subtopics associated with crowd analysis field.
After that, our contributions to crowd density analysis field are presented. It can be composed of two major parts:

1. In the first phase of this thesis, we focus on the problems related to the characterization of the crowd density by addressing people counting, density level estimation, and crowd motion segmentation using low level features.

    - In **Chapter 3**, we propose a novel solution of people counting problem, where only a frame-wise normalized feature is used in the regression step. To achieve this goal, distance and crowd density cues are explored. The first cue is employed to address the problem of perspective distortions, whereas, the second cue is used as crowd feature to detect and to measure the overlap between individuals. This solution is illustrated within two proposed approaches.
    - In **Chapter 4**, we handle the problem of crowd level classification. In particular, our research study is focused on the descriptive power of LBP features, and the impact of subspace learning on LBP features. In addition, we propose an alternative solution to multi-class SVM that maintains low computing complexity.
    - In **Chapter 5**, we propose a spatio-temporal model of crowd density using feature tracks as observations of a probabilistic crowd function. This measure has the advantage of providing local information of the crowd density compared to the other commonly used forms (i.e. number of people and crowd level). That is why, it will be further used in the applications presented in the second part of this thesis.

2. In the second part of the thesis, we demonstrate how a prior estimation of crowd density could provide valuable information and complement other applications in video surveillance. In particular, three applications are explored:

    - In **Chapter 6**, we present our proposed approach for enhancing human detection and tracking in crowded scenes which is based on incorporating crowd density and adding geometrical constraints to the detection and tracking process.
    - In **Chapter 7**, we propose a new application of crowd density measure in privacy context. The concept of context-aware privacy protection has recently emerged, as the required amount of privacy protection is deeply linked to the

context of the scene and the purpose of the monitoring activity. The effectiveness of the proposed contextualized privacy filters has been demonstrated by assessing the intelligibility vs. privacy trade-off for objective evaluation.

- In **Chapter 8**, we propose a novel approach to detect crowd change and to recognize crowd events. It is based on analyzing temporal and spatial distributions of persons using long-term trajectories within a sparse feature tracking framework to avoid the difficulties encountered by performing person tracking in crowded scenes.

In **Chapter 9**, we conclude about the presented works, highlight its limitations and suggest new research directions.

Publications

The featured list spans over all published and submitted documents of the author. Some of these publications appear in the Bibliography.

Journals

H. Fradi, and J. L. Dugelay, "Towards Crowd Density- Aware Video Surveillance Applications", Information Fusion Journal, Special Issue on *Intelligent Video Surveillance in Crowded Scenes*, Under Review.

H. Fradi, V. Eiselein, J. L. Dugelay, I. Keller, and T. Sikora, "Spatio-Temporal Crowd Density Model in a Human Detection and Tracking Framework", Signal Processing Journal: Image Communication, Under Review.

H. Fradi, and J. L. Dugelay, "Crowd Change Detection and Event Recognition by Modeling Attributes of Crowd Tracks", Signal, Image and Video Processing Journal, Special issue on *Semantic representations for social behavior analysis in video surveillance systems*, Under Review.

H. Fradi, X. Zhao, and J. L. Dugelay, "Estimating the Density of a Crowd using Computer Vision", IEEE COMSOC MMTC R-Letter, October 2013.

Conference Papers

H. Fradi, and J. L. Dugelay, "Sparse Feature Tracking for Crowd Change Detection and Event Recognition", in International Conference on Pattern Recognition (ICPR), August 2014.

H. Fradi, A. Melle and J. L. Dugelay, "Contextualized Privacy Filters in Video Surveillance Using Crowd Density Maps", in IEEE International Symposium on Multimedia (ISM), December 2013.

H. Fradi and J. L. Dugelay, "Crowd Density Map Estimation Based on Feature Tracks", in IEEE International workshop on Multimedia Signal Pressing (MMSP), September 2013.

H. Fradi and J. L. Dugelay, "A New MultiClass SVM algorithm and its application to crowd density analysis using LBP", in International Conference on Image Processing (ICIP),

September 2013.

V. Eiselein, H. Fradi, I. Keller, T. Sikora , J. L. Dugelay, "Enhancing Human Detection using Crowd Density Measures and an adaptive Correction Filter", in IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS), August 2013.

H. Fradi, X. Zhao, and J. L. Dugelay, "Crowd Density Analysis using Subspace Learning on Local Binary Pattern", in IEEE International Workshop on Advances in Automated Multimedia Surveillance for Public Safety (ICME workshops) July 2013

H. Fradi, V. Eiselein, I. Keller,J.-L. Dugelay, T. Sikora, "Crowd Context-Dependent Privacy Protection Filters", in 18th international conference on digital signal processing (DSP), July 2013.

H. Fradi and J. L. Dugelay, Low Level crowd Analysis using Frame-Wise Normalized Feature for People Counting", in IEEE International Workshop on Information Forensics and Security (WIFS), December 2012.

H. Fradi and J. L. Dugelay, "People Counting System in Crowded Scenes Based on Feature Regression", in European Signal Processing Conference (EUSIPCO), August 2012.

H. Fradi and J. L. Dugelay, "Robust foreground Segmentation using Improved Gaussian Mixture Model and Optical Flow", in International Conference on Informatics, Electronics Vision (ICIEV), May 2012.

H. Fradi and J. L. Dugelay, "Improved Depth Map Estimation in Stereo Vision", in SPIE, Electronic Imaging Conference on 3D Image Processing (3DIP) and Applications, January 2011.

CHAPTER 2

Video Surveillance Systems and Crowd Analysis

2.1 Introduction

In this Chapter, we first describe the common capabilities of automated surveillance systems. Then, we outline the challenges associated with crowd analysis field. Afterwards, we review the recent studies that have been proposed in this field. In particular, three main representative set of subtopics are introduced and briefly reviewed. More detailed review of the related works of each subtopic is given later per chapter, to provide a deep foundation on which our contribution in that chapter is based.

2.2 Automated Surveillance Systems

The past few decades have witnessed a widespread growth in the adoption of video surveillance systems mainly with the increasing performance of the cameras while their prices are dropping. Nowadays, video surveillance has become a key technology for modern society. The utilization of closed-circuit television systems (CCTV) has grown at an ever increasing rate and is becoming ubiquitous in almost all public areas to monitor individuals in airports, subways systems, sporting events and many other public facilities.
In traditional surveillance systems, the cameras are constantly monitored by a human operator which makes the effectiveness and the response of any surveillance system widely dependent on the vigilance of the person monitoring the camera. Besides, with the increasing number of cameras and the widespread areas under surveillance, this task is becoming more difficult, if it is not impossible.

To handle the limitations of traditional surveillance methods, a significant effort has been devoted in computer vision and artificial intelligence community to develop automated systems for monitoring persons, objects, and vehicles. The goal of these automated visual surveillance systems is to reduce the burdensome task done by video operators by giving a description of what is happening in a monitoring area and to consequently enable taking appropriate decision based on video footage. This description varies according to the context and the area being monitored, for instance, a survey in [105] highlights that to detect congestion in some areas has high priority for a public transport surveillance system.

In this context, there have been significant advances in automated visual surveillance systems [43, 21, 127, 113]. Nowadays, a modern surveillance system is expected to not only perform basic object detection and tracking, but also to provide a higher level interpretation of object behaviors. This could include several applications such as abnormal event detection, main traffic trends analysis, and improving object detection and tracking.
Given a video, a typical pipeline of video surveillance is composed of the following main steps: (1) Detection of objects to find areas of interest in the video. (2) Tracking of these objects from frame to frame to join these detections into records of a single object. (3) These records are further analyzed to detect the type of the object (to categorize car, person, bicycle, and so on) or its identity. (4) Analysis of the behavior of these objects to generate alerts when unusual behaviors are observed.
A typical pipeline of automated surveillance systems is shown in Figure 2.1.

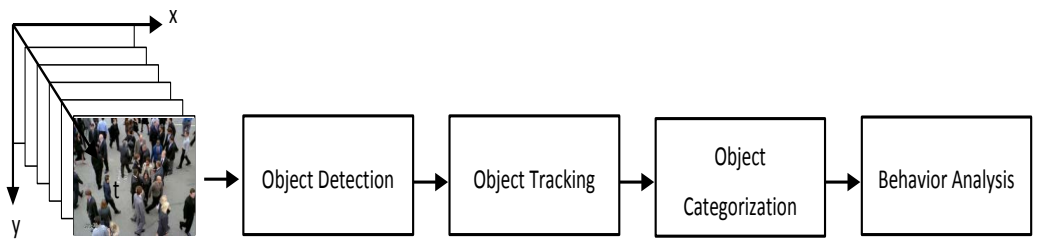


Figure 2.1: *Basic processing pipeline of current automated surveillance systems*

To sum up, an automated surveillance system has to be able to detect moving objects once they appear in the field of view of the camera, to track them over time, to classify them into different categories, and to recognize their activities and behaviors. Each of these subproblems has its challenges [127, 31], more details about that are given in the following sections:

2.2.1 Detection of interesting objects

The first step of automated surveillance systems is the detection of interesting objects in the camera's filed of view. This is a fundamental process at the core of any automatic video analysis. Usually, only moving objects are of interest, whereas static objects in the scene are not, which recasts object detection problem as the detection of motion. Practically, this detection can be performed depending on the camera; if pan-tilt-zoom (PTZ) cameras are used, once the camera moves, a change will occur in the whole image. Therefore, in this case, only techniques such as trained object detectors can be applied. However, the majority of video surveillance systems assume that the cameras are static. In this case, object detection can be performed by building a representation of the scene called back-

ground model. By comparing each incoming frame to this model, differences are flagged as moving objects. Many studies carried out this analysis on each pixel of the image independently [35]. A common approach to perform that is the work of Stauffer and Grimson [120], which consists of modeling each pixel as a mixture of Gaussians and using an on-line approximation for updating the model. Thanks to its ability to model various background distributions, this method showed a substantial progress to handle complex scenes. Therefore, until nowadays, GMM based background subtraction is considered as baseline method and it has become the basis for a large number of variations, for example, by performing shadow removal, or considering groups of pixels or texture [131, 51, 139].

Usually foreground regions that are detected and distinguished from the background model are processed by some morphological operations and connected components for further analysis. Other detection methods employ trained detectors to detect objects of a specific category. To perform that, object detectors are trained on databases such as pedestrians, and vehicles in order to detect instances of the object class in question in individual frames separately.

Object detections may be the goal of some particular applications such as the surveillance of secure areas where there should be no activity at all, or for minimizing video storage by capturing videos at low frame rate except when there is motion. However, many other surveillance systems require the tracking of these detections over time for further processing.

2.2.2 Object Tracking

Once interesting objects are detected, the following step consists of recording their movement over time to estimate trajectories. Tracking problem is based on aggregating multiple observations of a particular object into a track to enable the analysis of object's behavior. At any time step, it is formulated as a data-association problem, where new observations have to be assigned to tracks that represent the previous observations of some objects. This task is simple if the object is continuously observable and its appearance does not change over time. In this case, the problem is easy to be addressed because the observations of an object are similar. However, this is not almost the case since usually people undergo a change in their shape when they move, and their motion is not constant. Besides, there is usually discontinuity in the observation of the object, for instance, when another object cross in front of it. In such cases, the problem becomes more difficult, and to solve these issues more sophisticated algorithms are employed [133]. These include template trackers [50, 108], histogram-based trackers such as Mean Shift [22] and tracking based on contours detection, or color histograms [27]. More complex tracking problems can be solved using particle filters, and BraMBLe [59].

2.2.3 Object Categorization

After performing object tracking, a single track that corresponds to a single physical object (or a group of objects moving together) is built by associating multiple observations over the time. The next step consists of categorizing this object by providing its type or identity. In video surveillance systems, different objects could be observed such as a person, a group of people (if they are moving together), a vehicle, a bicycle and so on. Labeling the detected objects could provide valuable information for searches and automatic analysis, also it is important to recognize their activities. To perform that, usually, video surveillance systems have a set of predefined categories to figure out for example, people and vehicles [25], also to distinguish between different vehicles (if it is required) such car, and bus. These can be performed using shape information from a single or multiple images. If more rich information about the object and enough data are provided, this task could exceed type detection to perform object identification for example by face or gait for person recognition or by reading the license plate of a car.

2.2.4 Behavior Analysis

The task of behavior understanding consists of providing a high-level description of the actions and the interactions of and among the objects. This could include the detection if an object entered a certain area, the analysis of whether a particular action was carried out such as the detection of congestion, abandoned luggage [122], or act of aggression. Usually, the established patterns of activity to characterize the behavior are compared to observed normal behaviors, to detect if any abnormal behavior occurs. Depending on its degree of security threat, the outcome of a detected event should be configurable in a system from being recorded and stored in a database, to the automatic triggering of an alarm.

Much effort has been devoted on studying these problems in typical surveillance scenarios containing low density of persons. However, in high-density scenes these approaches are of limited success because it is almost impossible to delineate an object, to track it, and to analyze its behavior in a crowd. This inadequacy to deal with crowded scenes is a big problem because such situations often occur in practice (i.e in demonstrations, gathering in public spaces like markets, train stations, and airports) and are of big interest because it could lead to dangerous problems. Therefore, more recently, a number of studies have begun to focus on the analysis of high-density scenes.

2.3 Crowd Analysis

In this Section, we first introduce the main characteristics of crowded scenes. Next, we present the major subtopics of this field with a review of the recent studies that have been proposed to address the various challenges related to the analysis of crowded scenes.

Figure 2.2: Examples of high density scenes

Visual analysis of high density scenes (as those shown in Figure 2.2) is challenging compared to scenes with few people because of many reasons, mainly four points can be identified: Firstly, due to the large number of pedestrians within extremely crowded scenes, the size of an object is usually small in crowds. Secondly, the number of pixels of an object decreases with a higher density due to the occlusions caused by inter-object interactions. That substantially affects the appearance of the objects in video sequences because only some parts of each individual's body are visible. Thirdly, constant interactions among individuals in the crowd makes it hard to discern them from each other. Finally and as the most difficult problem, full occlusions that may occur (sometimes for long time) by other objects in the scene or by other targets.

Because of all these factors, the automated video surveillance pipeline described above has some difficulties to be applied in crowded scenes [114]. Actually, this type of video analysis has limited application to sparse and medium dense scenes. Thus, as the density of people increases in the scene, a substantial deterioration in the performance of object detection, tracking, and behavior analysis is observed. In the following, we review the recent studies linked to crowd analysis field [102, 60, 134]. These studies could be grouped into three major representative set of problems which are: (1) estimating the density of people in a crowd (2) detection and tracking of people in crowded scenes (3) crowd behaviors modeling and anomaly (or change) detection.

2.3.1 Crowd density estimation

An important problem in crowd analysis that has been studied in a number of works is crowd density estimation. Intuitively, different crowd density should receive different levels of attention. The objective of the related works that focus on this problem is either to provide an estimation of the crowd level, or to count the number of pedestrians.

The taxonomy of methods that perform crowd density estimation in the form of person counting embodies two paradigms: detection-based (direct) and regression-based (also called map-based or indirect) methods. The first paradigm consists of aggregating person counts from local object detectors. Once the object detector is applied, localizations of all person instances are given. Having obtained that, person counts can proceed in straightforward manner. By applying these methods, the count is not affected as long as people are correctly segmented. But, the difficulty is that detecting people is by itself a complex task. Detection-based counting can generate accurate estimation in low dense scenes, however they face some difficulties in high crowded scenes because of occlusions. This problem has been partially addressed by adopting part-based detectors [130], or by detecting either only heads [74, 125] or the Ω-shape formed by heads and shoulders [73]. These attempts to mitigate occlusions could be effective in medium crowd scenes, however, they are not applicable in very crowd scenes which are of primary interest for people counting.

Since analyzing crowded scenes still remain challenging (because of the spatial overlaps that makes delineating people difficult), most of the recent works bypass the task of detecting people and instead focus on extracting a set of low level image features. This paradigm of counting methods is based on regression to learn the relationship between the set of extracted features and the number of persons [98]. Once trained, an estimate for object counts can be obtained from the values of the extracted features. In this context, intensive study has been conducted by employing different features. Some of them are features of foreground pixels (e.g. total area, textures and edge count [15], [28], [67], [83]) and the others are based on measurements of interest points (e.g. corner points [3] and SURF features [23]). Also, this problem has been addressed by applying different regression functions (e.g. linear in [3] and [92], ε-SVR regressor and ANFIS in [2], Bayesian Poisson in [18] and Gaussian Process regression in [15]) to select the one fitting the features. This extensive study varying the features or the trainable function is caused by the fact that the features deviate from the perfect case where the number of persons is simply proportional to the features. Therefore, instead of training more features and testing different regression functions, we are interested in revealing the factors that affect the relationship between the features and the number of persons. More details about the related works to people counting problem and our proposed approach to handle that can be found in Chapter 3.

The estimation of person densities can be also in the form of crowd level, which is defined according to the number of persons per square meter within a given region of a

video. In this context, the definition from Polus *et al.* in [94] of different crowd levels is widely employed. It provides a clear definition of level of services from free flow to jammed flow according to a density metric defined as the number of persons per unit area. Learning to infer such crowd level within any region of known area could be formulated as a classification problem between a set of features and the different crowd levels. Based on the assumption that high density crowd has fine patterns of texture, whereas, images of low density have coarse patterns of texture [83], many texture features have been proposed to address the problem of crowd density estimation such as: Gray Level Co-occurrence Matrix (GLCM) [83, 65], Gradient Orientation Co-occurrence Matrix (GOCM) [80] and wavelet [84]. Among these features, GLCM is probably the most frequently used, from which usually 4 statistical properties are selected (contrast, homogeneity, energy, and entropy).

These statistical texture features have the limitation of giving a global information for the entire image. Also, these features could deal with occlusions that exist in crowded scenes only to some extent. As a result, the use of local texture features, especially some variants of LBP [91], has been an active topic of recent research (e.g. Dual-Histogram LBP in [78], spatio-temporal LBP in [132], GLCM on LBP image in [128], and an improved uniform LBP in [88]). These methods generally perform crowd density level classification directly using the high dimensional LBP-based feature vector, which might contain components irrelevant to crowd density. Also the use of the whole feature vector without a feature selection process could lead to unsatisfactory classification performances. Therefore, compared to the previous proposals based on applying different variations of LBP, in our study, we focus on the feature selection step by applying dimensionality reduction on the feature space. More details about the related works and the proposed approach for crowd density estimation can be found in Chapter 4.

2.3.2 Detection and Tracking in crowded scenes

Automatic detection and tracking of people in video data is a common task in the research area of video analysis and its results lay the foundations of a wide range of applications such as video surveillance, behavior modeling, security applications, traffic control, and mugging detection. Many tracking algorithms use the “Tracking-by-detection" paradigm which involves the application of a detection algorithm in individual frames and then estimates the tracks of different objects by associating the previously computed set of detections across frames. Tracking methods based on these techniques are manifold and include e.g. graph-based approaches ([52], [93]), particle filtering frameworks ([11]) and methods using Random Finite Sets ([34]).

Although there are different approaches to the tracking problem, all of them rely on efficient detectors which have to identify the position of persons in the scene while minimizing false detections (clutter) in areas without people. Techniques based on background subtraction such as [38] are widely applied thanks to their simplicity and effectiveness but

are limited to scenes with few and easily perceptible constituents. Generally, conventional tracking algorithms that focus on one particular object in the scene have some difficulties to deal with an unknown number of targets and the interactions among them in multi-target tracking problem. The application of these *object-centric* methods on videos containing dense crowds is therefore even more challenging and more issues could be encountered in such cases.

Crowded scenes exhibit some particular characteristics rendering the problem of multi-target tracking difficult. Targets are often occluded by other objects in the scene or by other targets which makes it difficult to distinguish one specific person from the others. Also, the size of a target in crowds is usually small which affects its appearance in video sequence. The aforementioned factors contribute to the loss of observation of the target objects in crowded videos. These challenges are added to the classical difficulties hampering any tracking algorithm such as: changes in the appearance of targets related to the camera view field, the discontinuity of trajectories when the target exits the field of view and re-appears later again, cluttered background, and similar appearance of some objects in the scene. Because of all these issues, human detection or tracking paradigms fail in such scenarios.

The problem of tracking in crowds has been studied in many works which attempt to perform that in scenes of medium-to-high density from monocular video sequence [5, 48, 75, 70, 11, 136] or recorded from multiple camera configurations [66, 42]. In medium crowded scenes, multi-target tracking could be performed by applying tracking-by-detection [11, 70]. Whereas, in extremely crowded scenes another category of methods has been recently proposed. It consists of learning motion patterns in order to constraint the tracking problem. For instance, in [5], global motion patterns are learned and participants of the crowd are assumed to follow a similar pattern. Rodriguez *et al.* [100] extend this approach to cope with multimodal crowd behaviors by studying overlapping motion patterns. These solutions are not suitable for tracking objects whose movements are not conform to the global motion patterns. Besides, these methods operate in off-line mode, they require the availability of the entire test sequence. Also, the learned patterns are tied to a particular scene.

Similarly, crowd density measures are employed in the literature to enhance person detection in crowded scenes. For instance, in [55], the number of persons is introduced as prior information to the detection step which is formulated as a clustering problem with a known number of clusters. But counting people is by itself a complex task in presence of crowds and occlusions. Besides, using the number of people as a crowd measure has the limitation of discarding local information about the crowd. Therefore, in [101], the authors investigated the idea of integrating a local crowd density measure in the detection and tracking process. Using an energy formulation, better results are obtained compared to the baseline method [40]. Despite the good results of this method, it includes some weakeness and leaves some rooms for improvements. For instance, the authors use the confidence scores from person detection as input to the density estimation which does

not introduce complimentary information into the process. In addition, a learning step with a given set of human-annotated ground truth detections is required, which makes the system not fully automatic. In contrast to the previous work, we intend to demonstrate the effectiveness of an automatic crowd description provided by crowd density maps in order to enhance human detection and tracking results. More details about the related works and the proposed approach for person detection and tracking in crowded scenes can be found in Chapter 6.

2.3.3 Crowd change modeling, detection and event recognition

Crowd behavior analysis has recently attracted research attention. This problem covers different subproblems such as crowd change or anomaly detection [12, 29, 86, 58, 118, 19, 12], and crowd event recognition [106, 47, 3, 16, 64, 135, 36]. The goal is to automatically detect changes and to recognize crowd events in video sequences. Usually the activity process in video sequence can be categorized into three main steps: (1) detection, (2) tracking, and (3) event recognition [47]. Given the difficulties encountered by analyzing crowded scenes, usually, research works related to crowd event recognition bypass the detection and the tracking of individuals in the scenes. Instead, some works focus on detecting and tracking local features [58, 19, 106, 3], or particles [86, 64, 135]. The extracted local features (points of interest) are employed to represent the individuals present in the scene. In this case motion patterns that have to be associated to individuals are assigned to the local features. By this way, tracking of individuals in crowds which is a daunting task is avoided. Likewise, alternative solutions that operate on particles tracking, observe that when persons are densely crowded, individual movement is restricted, thus, they consider members of the crowd as granular particles. Then, these methods proceed by putting a grid of particles over the image frame and moving them with the flow field. Other methods operate on foreground masks [29, 17, 12] by considering these foreground areas as regions of interest, denoted as *activity area* in [12].

In general, these methods aim at detecting and categorizing crowd events using motion information. This latter could correspond to normal (frequent) behavior or abnormal (unusual) behaviors. That is why, a general approach consists of modeling normal crowd behaviors, then abnormal behaviors are detected once a deviation from the normal behavior is observed. In [58], Ihaddadene *et al.* propose to detect sudden change and abnormal motion variations using motion heat maps and optical flow. The proposed approach is based on computing points of interest in the regions of interest (masks that correspond to areas of the built motion heat map). Then, the variations of motion are used to detect abnormal events. For this purpose, an entropy measure that characterizes how much the optical flow vectors are organized, or cluttered in the frame is defined in terms of a set of statistical measures using a pre-defined threshold. Another study that addressed the problem of abnormal crowd event detection is the social force model proposed by Mehran *et al.* in [86]. The

method is based on putting a grid of particles over the image frame and moving them with flow field computed from the optical flow. Then, to extract interaction forces, the social force is computed on moving particles. Finally, the interaction forces are used to model the normal behaviors using a bag-of-words representation and the ongoing crowd behaviors are determined through the change of interaction forces in time.

While most of the existing works in this field rely only on motion information, we consider that local density is also an important cue for early detection of crowd event and it could complement crowd dynamics information. Therefore, we intend to prove the effectiveness of implying density estimation in such high level applications since the risk of dangerous events increases when a large number of persons is involved. More details about the related works to crowd change detection and event recognition and our proposed approaches to this field can be found in Chapter 8.

2.4 Conclusion

In this Chapter, we described the significant progress achieved in video surveillance field from simple CCTV systems that allow a video operator to monitor different locations at the same time, to automatic surveillance systems that analyze videos which enables detection, tracking and behavior analysis. Then, in the second part, we summarized the challenges and the problems encountered by the applications of these tasks in crowded scenes. Afterwards, we reviewed the recent studies in crowd analysis field that cover different aspects such as the estimation of crowd density, detecting and tracking of individuals in crowded scenes, and modeling collective crowd events and behaviors.

Part I

Low Level Features Analysis for Crowd Density Estimation

CHAPTER 3

People Counting Using Frame-Wise Normalized Feature

3.1 Introduction

People counting has emerged as an increasingly important and dedicated problem in crowd analysis field over recent years. In particular, significant progress has been made in this field using features regression. In this context, perspective distortions have been frequently studied, however, crowded scenes remain particularly challenging and could deeply affect the count because of the partial occlusions that occur between individuals. To address these challenges, we propose a novel people counting method where a perspective normalization and a crowd measure-informed density are introduced into a single frame-wise normalized feature. Afterwards, the correspondence between this feature and the number of persons is learned by Gaussian Process regression. Two illustrations of this method have been proposed. The first one is based on measurements of interest points, where SIFT interest points are extracted and tracked. The second approach harness the advantages of incorporating an uniform motion model into Gaussian Mixture Model (GMM) background subtraction to obtain high accurate foreground segmentation. Then, the count is based on measurements of foreground pixels.

3.2 Related Works

Significant progress has been achieved in the field of people counting using regression-based (indirect) methods. This category of methods has become a complementary solution where it is nearly impossible to isolate and to count each person in crowded areas. In this section, a brief description of this category of methods is provided, along with some representative approaches. Unlike detection-based counting paradigm which consists of providing the number of persons and their locations simultaneously, regression-based counting paradigm consists of estimating the number of persons from various features. This paradigm of counting methods is more efficient, since it is easier to detect features than to detect persons. For this purpose, many features of foreground pixels (e.g. total area, textures and edge count [15], [28], [67], [83]) and also features based on measurements of interest points (e.g. corner points [3] and SURF features [23]) are employed in counting

methods. Then, to perform the counting, a regression function has to be applied. It is required to learn the relationship between features and number of persons.

More in details, Hou *et al.* [54] addressed this problem by using a neural network to map the foreground pixels to the number of persons. In this work, the foreground pixels are extracted by subtracting each frame from a learned statistical background model. In [3], Albiol *et al.* proposed to use Harris corner points as features. Then, the count is performed by assuming a direct proportional relation between the number of corner points and the number of persons. This method has shown good performance using PETS dataset, whereas, its application is limited because it does not consider the difference between the perceived size of persons at different distances from the camera and with different densities as well. These limitations were not revealed in the PETS contest since only videos characterized by short depth range and trivial occlusions were required for the tests.
Differently from the two aforementioned works, some other research take into account the impacts of perspective distortions. To handle that, different techniques have been investigated. For instance, in [92], this problem is addressed by weighting foreground pixels according to geometric information. In [77], Ma *et al.* propose a geometric correction to bring all the objects at different distances to the same scale. Then, a linear relationship is established between the number of foreground pixels and the number of persons.

While different techniques have been proposed to address the problem of perspective distortions, only few attempts have been done to handle the problem of occlusion that exists in the crowd and could deeply affect the count. In [15], Chan *et al.* applied a mixture of dynamic texture to segment crowd video. Then, for each crowd segment, 28 features are extracted and weighted by applying a perspective map to each image location according to its approximate size in the real scene. These features varies among geometric, edges and texture. The reason behind using these various features is to be able to better interpret the image contents mainly to implicitly have a deep idea about the level of the crowd. Also, in [23], both of perspective and crowd problems have been addressed by applying a clustering algorithm to partition different groups of persons. Then, the distance from the camera is computed using an Inverse Perspective Mapping (IPM) and the density of each cluster is obtained as the ratio between the number of the detected points and the area of the bounding box. Although this method [23] proposes to deal with two major problems that usually affect the results of regression-based methods, it still suffers from many limitations and leaves rooms for improvements. One of the drawbacks is that it requires three parameters (number of detected points, distance, and density) for each cluster separately, which is a heavy annotation task. More details about the limitations of this method will be discussed along the development of our proposed approach. Recently and differently from the previous works, in [119], an explicit estimation of the crowd levels is involved and the number of persons is estimated through a scaling factor which is learned for different levels of the crowd.

These solutions based on using various features (it reaches 28 features in Chan's method

[15]) or by involving an estimation of the crowd level to adjust the scaling factor (as in [119]), were applied to infer additional information about the frame contents. Also, this problem has been addressed by applying different regression functions to select the one fitting better the features. Ideally, the number of persons is simply proportional to the features, but some factors are affecting this relationship which leads to a deviation from the proportionality. Therefore, varying the features or the trainable functions are just applied as an implicit way to cope with this deviation and to infer more information about the frame contents.

The remainder of the Chapter is organized as follows: In Section 3.3, we introduce our approach for people counting based on frame-wise normalized feature. The regression step is presented in 3.4. The proposed approach is evaluated using PETS dataset and the experimental results are summarized in Section 3.5. Finally, we briefly conclude.

3.3 Frame-Wise Normalized Feature Extraction

To perform people counting, we follow the methods based on features regression. One major advantage by applying these methods is that they do not depend on intermediate steps of individual detection or tracking. Unlike the previous works which are based on varying the features or the trainable functions, in our study, we are more interested in revealing the factors that affect the relationship between the features and the number of persons. In particular, we intend to explore distance and crowd density cues. The first cue is employed to handle the problem of perspective distortions, whereas, the second cue is used as a crowd feature to detect and to measure the overlap between individuals. To achieve this goal, a perspective map normalization and a crowd measure are applied in order to compensate the variations in distance and in density. These two normalizations are introduced into a single frame-wise normalized feature. Our intuition behind this is to make the feature invariant to the aforementioned factors.
The two normalization are detailed as follows:

- **Perspective normalization:**
  The objective is to compensate for changes in number of extracted features due to perspective distortions. The effects of perspective can be simply explained by the fact that objects far away from the camera appear smaller than the closest ones. This makes any extracted feature from farther away persons account for a smaller portion compared to closer persons. This problem could be addressed by weighting each extracted feature according to a perspective map with assignment of larger weights for farther points in the scene.

  Similar to [15], we estimate the perspective map by linearly interpolating between the two extremes of the scene. First, the ground plane is marked. Then, the distance d_1 and d_2 of the two extreme lines are measured. After that, the difference between

the perceived height of persons in these two lines can be derived by manually calibrating two frames, where the center of a reference person belongs to the first line in the first frame while belonging to the second extreme line in the second frame. A weight of 1 is assigned to pixels on the first line, and the pixels on the second line are weighted by $\frac{h_1 * d_1}{h_2 * d_2}$, where h_1 and h_2 denote the two heights of the reference person in the two frames. A linear interpolation is applied to compute the remaining weights between the two extreme lines. As a result, different weights W_p are assigned according to the y-coordinate.

- **Density normalization:**
  In addition to perspective distortions, the extracted features are also extremely sensitive to the "crowdedness" (level of the crowd density). When people are closer to each other, less points are extracted due to the partial occlusions that occur. Thus, we intend to estimate the density of people by measuring how close local features are. Moreover, we aim to handle this problem of variations in crowd density without involving an explicit estimation of the crowd level. This inspires us to search for a way that can directly weight the feature. Therefore, we propose to synthesize local features for a global density. Then, we aim at formulating a weighting function by using the density of local features as a crowd measure. In particular, our goal is to weight the extracted features by inflating its values in high crowd situations, while reducing them in low crowd situations. Thereby, we use the estimated density values d_k, $k = 1...M$ from the training set, where M is the total number of frames in the video sequences used for training. And we define the weight function of a new testing sample i as:

$$W_d(i) = \frac{d_i - \mu}{\sigma_{max}} + 1 \tag{3.1}$$

  Where $\mu = \frac{1}{M} \sum_{k=1}^{M} d_k$ and σ_{max} is the maximum of standard deviation values σ_k. This weight function ensures crowd normalization; it is achieved by setting W_d to 1 if the crowd is medium ($d_i = \mu$), $1 < W_d \leq 2$ if the crowd is high, and $0 \leq W_d < 1$ otherwise. In any case, the upper bound of W_d is equal to 2 and the lower bound is equal to 0.

In the following paragraphs, our proposed approaches for people counting based on measurements of interest points and based on measurements of foreground pixels are presented, respectively, in 3.3.1 and in 3.3.2.

3.3.1 Based on measurements of interest points

The methods based on using interest points [23, 3] have the advantage of bypassing intermediate steps of foreground segmentation as used in [15]. In this section, our proposed approach for people counting based on measurements of interest points is presented [45].

First, SIFT interest points are extracted. Then, to filter out the static detected points, motion information is estimated. For this purpose, an efficient solution based on computing the optical flow with reduced weights near the borders since the expansion coefficients are less reliable there is proposed. Moreover, as mentioned before, in this study we explore distance and density cues in order to compensate the effects of perspective distortions and partial occlusions due to the crowd. These two factors were not taken into account in [3]. Also in [15], the effects of people density were not considered, however, 28 features from foreground pixels were devoted to infer the contents of each frame. Conte's method [23] is the only work that dealt with the two aforementioned factors, but the proposed approach is still problematic. Compared to [23], we propose to process the perspective normalization at row level which is more accurate than assigning one distance value to each group of persons. In addition, for density estimation, we apply density-based clustering which is better adapted for separating different groups of persons than the graph-based clustering proposed in [23]. Another problem is addressed in our study; it is the calculation of the area of each cluster. We apply α-shape technique which is more powerful than the bounding box proposed in [23]. This latter fails to define boundaries of a set of points by leaving large gaps which could amply deteriorate the estimated density. Added to that, one major contribution of our counting system compared to Chan's method is the formulation of a new weight function based on density estimation for crowd normalization.
An overview of our counting system modules and their interactions is shown in Figure 3.1. The remainder of this section describes each of these system components.

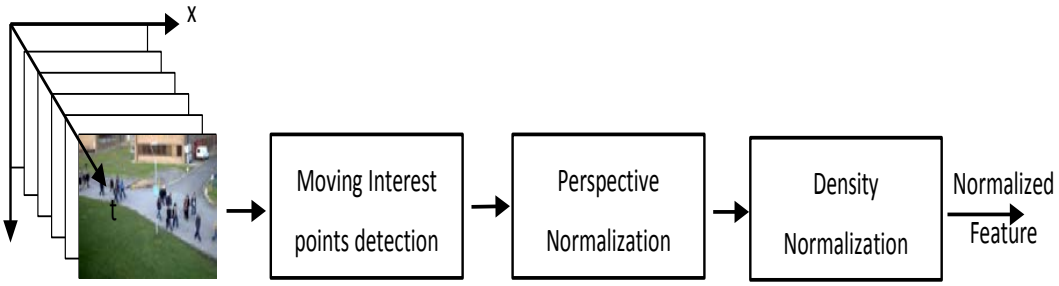


Figure 3.1: *Schematic for frame-wise normalized feature extraction based on measurements of interest points*

3.3.1.1 Detection of moving interest points

To infer the contents of each frame under analysis, only interest points are detected. For this purpose, we propose to use SIFT (scale-invariant descriptor) [76] where interest point locations are defined as maximum/minimum of the difference of Gaussians in scale-space. After that, motion information is associated to the detected interest points to distinguish between moving and static ones. By considering the same assumptions as in [23], the detected

interest points with non-null motion vector typically belong to persons. To perform that, we compute the optical flow using the method proposed in [39] which employs quadratic polynomial model to approximate each neighborhood of two consecutive frames. Then, the displacement fields are estimated from the polynomial expansion coefficients. This method has the advantage of reducing the errors near the borders by computing the polynomial expansions with certainty set to zero off the border and with a reduced weight for pixels close to the borders. Then, to take into account the effects of perspective distortions, the weights W_p (defined above) are assigned according to the y-coordinate of each interest point.
After perspective normalization, the number of moving SIFT points in each frame i under analysis is updated as follows:

$$FeatN_p^1(i) = \sum_{y=1}^{Y} W_p(y) * N_T(y) \tag{3.2}$$

Where $N_T(y)$ is the total number of moving points in the y^{th} row.

3.3.1.2 Estimation of the density of interest points for crowd measurement

To estimate the density of moving interest points, a clustering algorithm is applied first. It is required to distinguish between detected points belonging to different groups of persons. The most appropriate solution for this problem is density-based clustering, where clusters are identified according to the spatial density of the points. It also has the advantage of being flexible enough to discover clusters of arbitrary shape. After that, the boundaries of each cluster are defined using the α-shape technique.

Density-based clustering:

For density-based clustering, we apply DBSCAN (Density Based Spatial Clustering of Applications with Noise) [37]. This algorithm does not require any prior knowledge about the number and the shape of the clusters. Added to that, it fits well our requirements by adopting the concept of density-reachable to form the clusters with respect to $MinPts$ and Eps input parameters which denote, respectively, a threshold of points needed in a neighborhood and a neighborhood radius. Moreover, points which are not density-connected are labeled as noise. In Algorithm 1, we present a basic version of DBSCAN.

Density estimation:

The density is measured by computing the ratio between the number of moving interest points and the area covered by the clusters. For the area computation, we propose to delineate the boundaries of each cluster by α-shape [33] which is an accurate technique to extract the shape of a set points compared to the bounding box employed in [23], see Figure

Algorithm 1: DBSCAN

Input: $D\{x_i\}$: set of points, Eps : neighborhood radius, $MinPts$: threshold number of points needed in a neighborhood.
Output: Set of clusters, Noise

$id \leftarrow 0$
for $x_i \in D$ and x_i is UNCLASSIFIED **do**
  Mark x_i as CLASSIFIED
  $N \leftarrow$ Neighbors of $\{x_i \in D | d(x_i, x_j) \leq Eps, x_j \in D \setminus x_j\}$
  if size$(N) < MinPts$ **then**
    Mark x_i as Noise
  else
    $id \leftarrow id + 1$
    Add x_i to $Cluster_{id}$
    for all $x_p \in N$ **do**
      if x_p is UNCLASSIFIED **then**
        Mark x_p as CLASSIFIED
        $N' \leftarrow$ Neighbors of $\{x_p \in D | d(x_p, x_q) \leq Eps, x_q \in D \setminus x_p\}$
        if size $(N') > Minpts$ **then**
          $N \leftarrow N \cup N'$
        end if
      end if
      if $x_p \notin Cluster_{id}$ **then**
        Add x_p to $Cluster_{id}$
      end if
    end for
  end if
end for

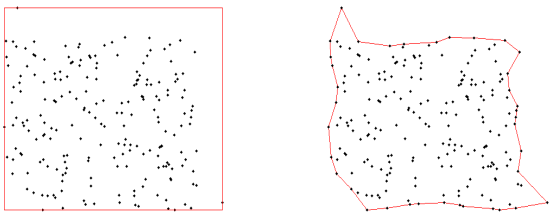
3.2 as an illustration of the differences between the two techniques.

α-shape technique has not only the advantage of closely following variations in the outer-edge but it also reveals the inner gaps. This technique is reliable to accurately estimate the density of clusters mainly together with the density-based clustering algorithm that picks out the clusters using the density relevance and filters out the noise.
Using the estimated density values, the effects of the crowd on the detected interest points are taken into account by computing the weight function defined in (3.1). Then, our proposed feature defined in (3.2) is again updated as follows:

$$FeatN^1_{p,d}(i) = W_d(i) * \sum_{y=1}^{Y} W_p(y) * N_T(y) \tag{3.3}$$

3.3.2 Based on measurements of foreground pixels

In this section, our proposed approach for people counting based on measurements of foreground pixels is presented [44]. Given the importance of foreground segmentation and its impact on the next steps, an efficient solution based on integrating GMM background subtraction with motion cue is employed [46]. Afterwards, only two holistic features are used:



(a) *Bounding box* (b) α-*shape*

Figure 3.2: *Bounding box and α-shapes of a corresponding set of points*

foreground pixel counts and corner density. The first holistic feature is weighted according to the estimated perspective map in order to compensate the effects of perspective distortions. We additionally explore density cue to handle partial occlusions due to the crowd. Under the assumption that images of low density crowd tend to present less dense corners compared to images of high density crowd, we propose to associate dense or sparse corners to the crowd size. For this purpose, Features From Accelerated Segment Test (FAST) are extracted and synthesized for global corner density. An overview of the feature extraction modules is shown in Figure 3.3.

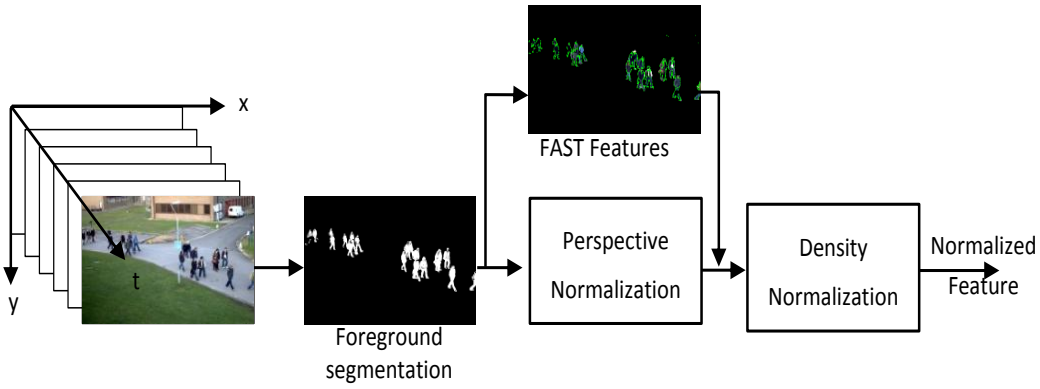


Figure 3.3: *Schematic for frame-wise normalized feature extraction based on measurements of foreground pixels*

3.3.2.1 Foreground segmentation

The first step of our proposed approach is to segment foreground entities. In this context, GMM background subtraction [120] has been widely employed. It is based on a probabilistic approach that achieves satisfactory performance to handle complex scenes thanks to its ability to model various background distributions. Therefore, until nowadays, GMM

based background subtraction is considered as baseline method and it has become the basis for a large number of extensions. Despite this, GMM includes some weakness. First, there is no consideration of spatial information. Second, the background model estimation step is problematic; the main difficulty is to decide which distributions of the mixture belong to the background, GMM assumes that the often occurring pixels are deemed to model the background which is not always true. Also, to adapt variations in the background (to maintain good precision), the detection rate is decreased. To overcome these limitations, we apply our proposed integration of GMM background subtraction with an uniform motion model [46]. For this, we use the improved adaptive GMM [138] which has the advantage of constantly updating not only the parameters of the Gaussians but also the number of the mixture components using the Dirichlet prior. The second cue of this method is motion information, it is obtained by computing the optical flow between each two adjacent frames [39]. The optical flow field is defined by its magnitude and its direction. The magnitude of motion is convoluted with the difference between each current frame and the mean of the background to get precise boundaries. After that, a measure of uniformity of motion is applied to distinguish different connected components with the same velocity and orientation of the optical flow. Finally, the labeling process is updated by favoring pixels moving together to be classified as foreground entities. The goal of this integration is to improve the detection rate of GMM and to avoid outliers caused by the optical flow as well. It could also add spatial and temporal coherence known that the labeling process using GMM is done only at pixel level, more details about our improved foreground segmentation method using GMM and motion cue are given in Appendix A.

After performing foreground segmentation, we note that only using the total number of foreground pixels is not enough to estimate the number of persons. The total number of foreground pixels in each frame i under analysis has to be updated by perspective normalization:

$$FeatN_p^2(i) = \sum_{y=1}^{Y} W_p(y) * FG_T(y) \tag{3.4}$$

Where $FG_T(y)$ is the total number of foreground pixels in the y^{th} row. Moreover, further enhancement of this feature is necessary to improve its invariance to crowd density, see next paragraph.

3.3.2.2 Corner density estimation for crowd measurement

In addition to perspective normalization, we also aim at making the feature defined in (3.4) invariant to crowd density. For this purpose, FAST local features are first extracted, then, they are synthesized for global corner density.

Local Features Extraction

For local features, we extract FAST [104] which is developed for corner detection in a fast

and a reliable way. It depends on wedge model style corner detection. Also, it uses machine learning techniques to automatically find optimal segment test heuristics. The segment test criterion considers 16 surrounding pixels of each corner candidate P. Then, P is labeled as corner if there exist n contiguous pixels that are all brighter or darker than the candidate pixel intensity. The reason behind applying FAST as local feature for crowd measurement is its ability to find small regions which are significantly different from their surrounding pixels.

After extracting FAST local features, the corner density is estimated by computing the ratio between the number of FAST corners and the number of foreground pixels. The objective of estimating that is to handle the problem of variations in crowd density. Using the corner density values, the weight function defined in (3.1) is computed and the feature defined in (3.4) is updated as follows:

$$FeatN^2_{p,d}(i) = W_d(i) * \sum_{y=1}^{Y} W_p(y) * FG_T(y) \tag{3.5}$$

3.4 Gaussian Process regression

The two proposed frame-wise features, based on measurements of interest points and of foreground pixels defined, respectively, in (3.3), and in (3.5) have been formulated to be invariant to perspective and to crowd density. This could ensure the linearity of the trainable function mapping the features to the number of persons. Fore more flexibility, we suggest to consider any eventual errors that could occur in the crowd segmentation or in any other step of our counting system. Therefore, we propose to use Gaussian Process (GP) regression which is well adopted for linear features with local non-linearities (more details about GP can be found in [96]).
The entire system architecture is illustrated in Figure 3.4.
In this chart, there are two flows: the training and the testing flows. In the training flow, the trainable function is learned from a set of labeled examples by using GP regression. Once the trainable function is estimated, the number of the persons could be predicted from the value of the proposed feature for each frame under analysis in the testing flow.

3.5 Experimental Results

In this section, we present the experimental results on the Performance Evaluation of Tracking and Surveillance (PETS) 2009 public dataset [41]. From this dataset, we are interested in the section used to assess *Person count and Density estimation* algorithms. Only 4 videos from the first view were tested in people counting contest held in PETS 2009. Since more tests under situations with important perspective distortions and occlusions are required to evaluate the counting system, we also employ other videos from the second view in our

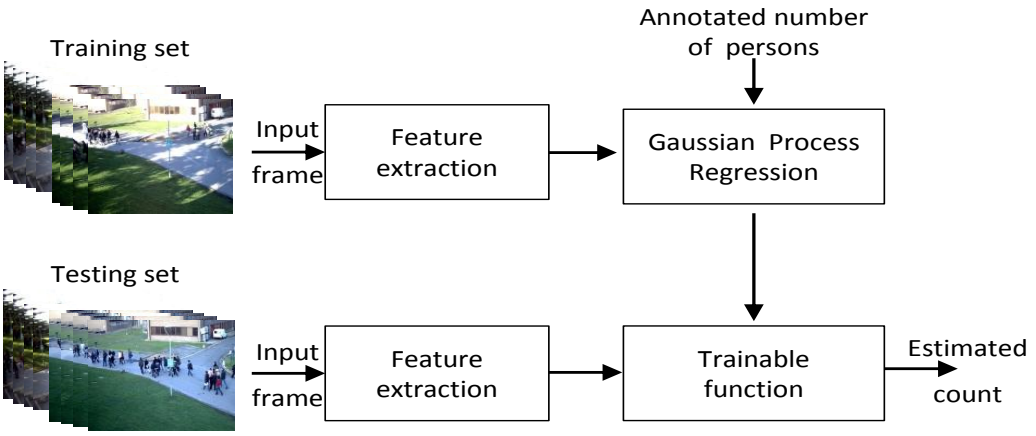


Figure 3.4: *Flowchart of people counting system*

experiments. The main characteristics of these videos are summarized in Table 3.1.

			Number of people	
Video Sequence	View	Length	Min	Max
S1.L1.13-57	1	221	5	34
S1.L1.13-59	1	241	3	26
S1.L2.14-06	1	201	0	43
S1.L3.14-17	1	91	6	41
S1.L1.13-57	2	221	8	46
S1.L2.14-06	2	201	3	46
S1.L2.14-31	2	131	10	43
S3.MF.12-43	2	108	1	7

Table 3.1: Characteristics of 8 sequences from the PETS 2009 dataset used for the counting experiments.

The ground-truth of the count is obtained by annotating the number of persons by hand in every 5^{th} frame. The count for the remaining frames is obtained using linear interpolation.

To compare the estimated number of persons to the ground truth, we calculate the Mean Absolute Error (MAE) and the Mean Relative Error (MRE) which are defined as:

$$MAE = \frac{1}{N}.\sum_{i=1}^{N}|E(i) - G(i)| \tag{3.6}$$

$$MRE = \frac{1}{N}.\sum_{i=1}^{N}\frac{|E(i) - G(i)|}{G(i)} \tag{3.7}$$

Where N is the total number of frames in a video sequence. E(i) and G(i) denote, respectively, the estimated and the ground-truth number of persons in the $i-$th frame. The MAE metric was used to compare the performance of the algorithms submitted to the PETS contest. But, the same error could be negligible if the number of persons is high. Therefore, in [23] the authors propose to also use the MRE metric, which relates the error to the number of the persons.

We start by evaluating the proposed approach based on measurements of interest points. For the comparisons, unfortunately, we are not able to compare our proposed method to Chan's method [15]. In fact, for their work [17] submitted to PETS 2009, only tests with videos from the first view were provided. Since, we are interested to test more challenging videos; our results are compared to those of Albiol and Conte methods [3, 23] reported in [23]. A summary of the counting results, with respect to the hand-annotated ground-truth, are given in Table 3.2.

Video Sequence	Albiol *et al.* [3]		Conte *et al.* [23]		Our approach	
	MAE	MRE	MAE	MRE	MAE	MRE
S1.L1.13-57 (View 1)	2.80	12.6%	1.92	8.7 %	1.38	7.10 %
S1.L1.13-59 (View 1)	3.86	24.9 %	2.24	17.3 %	2.25	15.02 %
S1.L2.14-06 (View 1)	5.14	26.1 %	4.66	20.5 %	4.58	21.75 %
S1.L3.14-17 (View 1)	2.64	14.0 %	1.75	9.2 %	1.54	8.99 %
S1.L1.13-57 (View 2)	29.45	106.0 %	11.76	30.0 %	3.64	11.67 %
S1.L2.14-06 (View 2)	32.24	122.5 %	18.03	43.0 %	6.87	18.30 %
S1.L2.14-31 (View 2)	34.09	99.7 %	5.64	18.8 %	2.53	10.93 %
S3.MF.12-43 (View 2)	12.34	311.9 %	0.63	18.8 %	2.20	40.31 %

Table 3.2: Quantitative evaluation of our proposed approach based on measurements of interest points compared to other regression-based methods

In this Table, it is shown a significant difference in the performance of Albiol's method [3] between the first and the second views. That could justify the incapability of this method to deal with challenging situations. Whereas, the method of [23] proposes to handle perspective distortions and density which are the two major problems that usually affect the results of regression-based methods. That could justify as well the better results of Conte's method compared to [3].

A comparison of our results with the results of [23] reveals the effectiveness of our proposed approach. As stated earlier, Conte's method [23] is the only work that dealt with

the two aforementioned factors, but this approach is still problematic as it is demonstrated in the results. One of the drawbacks of [23] is that it assigns one distance value to each group of persons which is less accurate than processing the perspective normalization at row level. It also includes other weakness such as the clustering algorithm which is not well adapted for separating different groups of persons, and the bounding box used to define the boundaries of interest points which fails to accurately delineate that by leaving large gaps. All these problems could amply deteriorate the estimated density value. It is also important to note that Conte's method requires three parameters (number of detected points, distance, and density) for each cluster separately, which is a burdensome annotation task. All these reasons could justify that our proposed approach outperforms the two others methods with respect to MAE and MRE metrics. In particular, the tests with S1.L1.13-57(2) and S1.L2.14-06(2) show the effects of the proposed crowd measure to compensate the underestimation of number of persons because of the dense crowd. Only for the video S3.MF.12-43 (2) (which has the lowest number of people), Conte's method gives better results than our proposed method. This result might be explained by the selected training set which did not include enough samples of frames containing few people.

Additionally, we access our proposed approach for people counting based on foreground measurements. The accuracy of the counting results is almost the same compared to those of the proposed approach based on measurements of interest points; MAE is about 25 for all the videos for the two proposed methods.
The interesting results of our proposed method could justify the effectiveness of the foreground segmentation method and the impact of the proposed normalizations. In fact, because of the complexity of the scene mainly in the second view, providing accurate results using foreground measurements is widely dependent on the first step which is the foreground segmentation. In this context, one of the problems that we faced using PETS dataset is the moving grass that occurs at several frames. GMM succeeds to handle this problem, but at the same time, adapting more variations in the background yields to a decline in the detection rate. Here comes the importance of applying the integration of GMM background subtraction with motion information into a single framework. By doing so, better segmentation of the scene into foreground and background entities is achieved and it is expected to bring a good performance to people counting. To justify these observations, precisely, to demonstrate the impact of the foreground segmentation step on the accuracy of people counting results, we compare MAE metric for the 8 videos (ordered by the same way as in Table 3.2) between applying the improved GMM [138] and applying our integration of the improved GMM with motion cue [46], see Figure 3.5. This comparison highlights an overall performance using [46]. Likewise, we prove the effectiveness of our proposed approach by showing that the two normalizations (perspective normalization and crowd density normalization) significantly enhance the accuracy of the counting results, see Figure 3.6.

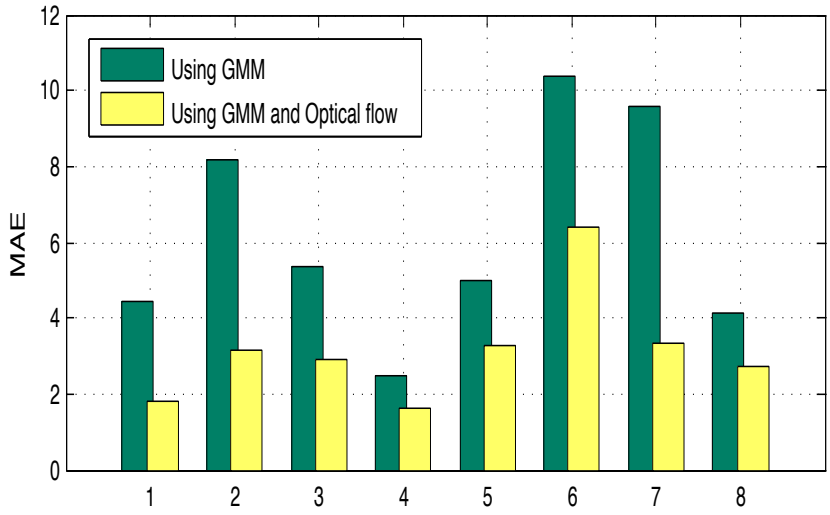


Figure 3.5: *Improvement made by using motion cue with GMM background subtraction*

3.6 Conclusion

In this chapter, we proposed a new concept for addressing people counting problem through two different approaches. It based on regressing a single frame-wise feature independent from variations of perspective and crowd density. Our contribution regarding the related works in people counting is discussed along the details of the proposed approaches. Also, experiments on PETS dataset demonstrate that our approaches achieve good results under situations of heavy occlusions and important perspective distortions. By means of comparisons with other existing regression-based methods, our results demonstrate the ability of our approaches to significantly improve the counting accuracy. Also, we show other experiments that highlight the role of the two normalizations and the integration of motion cue with GMM background subtraction as well.

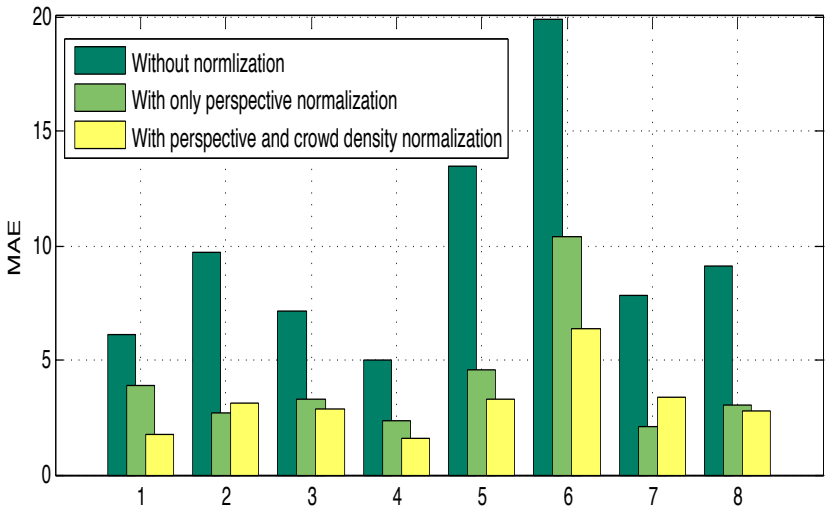


Figure 3.6: *Improvement made by normalizing the foreground pixels counts against perspective distortions and crowd density variations*

CHAPTER 4

Crowd Level Estimation using Texture Features Classification

4.1 Introduction

In addition to people counting problem, crowd level estimation is an important component in visual surveillance systems for crowd monitoring and management. In this Chapter, we propose a novel approach for crowd density estimation at patch level, where the size of each patch varies in a such way to compensate the effects of perspective distortions. The proposed approach consists of finding a low dimensional discriminative subspace in which same-density-level samples are projected close to each other while different-density-level samples are projected further apart. Specifically, Local Binary Pattern (LBP) feature vectors are projected into discriminant space using Linear Discriminant Analysis (LDA) over the Principal Component Analysis (PCA) subspace. This process is favorable for the later multi-class Support Vector Machine (SVM) classification step since the influence of feature components irrelevant to crowd density is minimized. In addition to the feature extraction block, an untapped potential to reduce the complexity of multiclass SVM problem is explored in this Chapter. Our alternative algorithm is based on automatic crowd judgments using relevance scores, which is less computationally demanding than one-vs-one and one-vs-rest multiclass SVM methods.

4.2 Related Works

In order to address the problem of crowd level estimation, many works have been proposed so far. In this context, the classification introduced by Polus [94] is commonly adopted, based on that, the crowd density is categorized into 5 levels: free, restricted, dense, very dense, and jammed flow. One of the key aspects of crowd density analysis is related to the extracted crowd features. Early attempts to handle this problem generally made use of texture features which are more frequently used than statistical pixel features (that are usually used for addressing people counting problem). Marana *et al.* [83] assume that high density crowd has fine patterns of texture, whereas, images of low density have coarse patterns of texture. Based on this assumption, many texture features have been proposed such as: GLCM [83, 65], GOCM [80] and wavelet [84]. Recently, the use of local texture

features has been an active topic, especially some variants of LBP [91]

For instance, in [78], an extension of the original LBP is used. Specifically, LBP is used in blocks, then, Dual-Histogram LBP (DH-LBP) is computed. By combining merits of both, the proposed advanced LBP (ALBP) is applied for solving the problem of crowd density estimation. In [132], the dynamic texture of the walking crowd is used by extracting a sparse spatio-temporal local binary pattern (SST-LBP) feature. Afterwards, the statistical property of SST-LBP is used to describe the crowd feature. Finally, the crowd features are classified into a range of density levels by adopting Support Vector Machine. In [128], the authors propose a novel texture descriptor called LBP Co-occurrence Matrix (LBPCM) which consists of computing GLCM on LBP image instead of the original gray image. LBPCM is constructed from several overlapping cells in an image block, and is classified into different crowd density levels. In this work, the experimental results demonstrate that concatenating LBPCM on gray and gradient images gives better results than doing that separately. Finally, in [88] a crowd density estimation approach using histogram model classification is proposed, where the histogram model is based on an improved uniform local binary pattern. The advantages of using this improved LBP are that the pattern features are intensity and rotational invariant. The experimental results demonstrate better results compared to the original LBP as well.

The methods mentioned above generally perform crowd density level classification directly using the high dimensional LBP-based feature vector, which might incur at least two problems: first, the high dimensional feature vector increases the computation time; second and more important, these high dimensional feature vectors generally contain components irrelevant to crowd density, and the use of the whole feature vector without any feature selection process could lead to unsatisfactory classification performances. Besides, the application of Polus definition of different levels of the crowd [94] includes some shortages: some of the related works estimate crowd level on patches, however, only one work [79] considers the effects of perspective distortions on the patch size. And none of these works attempt to estimate the real size of the frame or the sub-regions within frame, which leads to an unreliable usage of the definition of different crowd levels. Added to that, there is still untapped potential to reduce the complexity of the multi-classification problem while assigning texture features to crowd levels.

In the following, our proposed approach for crowd density estimation is presented. First, we introduce patch level analysis which involves the estimation of patch size in the real-world coordinates with incorporation of the effects of perspective distortions on the patch size, see Section 4.3. Then, to infer the contents of each image patch under analysis, texture features are extracted using subspace learning (or dimensionality reduction) on block-based LBP instead of using raw LBP. Specifically, the feature vectors are projected into discriminant space using LDA over the PCA subspace, see Section 4.4. Afterwards, the extracted features are classified into different crowd density levels by applying multiclass SVM, see Section 4.5.

4.3 Patch-Level Analysis

We propose to perform crowd density estimation at frame sub-regions, which is commonly referred as *patch level*. Crowd density at patch level is more appropriate than at frame level, since it enables both the detection and the location of potential crowded areas within the whole frame. Actually, in many video surveillance applications and for security reasons, not only the estimation of the crowd level is required, but also the location of the crowd within the whole frame. Moreover, estimating the crowd density at image patches enables to work within regions of interest. In fact, more interest is usually given to the prediction of the crowd level in some specific areas compared to others, such as in the walkways.

To assign image patches to crowd density levels, the first difficulty underlined in our work concerns the implementation of crowd levels definition introduced by Polus [94]. It consists of defining 5 crowd levels according to the range of density. This definition has been widely used for crowd density estimation, but, the estimation of the real size (of image, or image blocks) is usually neglected in previous works. In our proposed approach, we use the camera calibration parameters [124] to transform the image coordinates to the real-world coordinates, from which we can estimate the real size of any RoI within a frame. At this stage, we also take into account the effects of perspective distortions on patch size. Similar to the real size estimation, this problem is also not studied in the literature except in [79], where an approximation of the perspective map is made by linearly interpolating the two extreme lines of the scene. In our approach, to use only one definition of crowd levels under different locations within the whole frame, the effects of perspective distortions are compensated on the patch sizes in such way that all the extracted patches correspond to a similar size in the real-world coordinates.

4.4 Subspace Learning on Local Binary Pattern

4.4.1 Block-based Local Binary Pattern extraction and histogram sequence normalization

Recently, LBP [91] has aroused increasing interest in many applications of image processing and computer vision, in particular, it has been extensively studied and thus, related to the field of face recognition. Likewise, substantial progress has been achieved over the last years in crowd density analysis using LBP. The advantage of using LBP as feature extractor is that it is a powerful descriptor that characterizes the structure of the local image texture which is highly relevant to the crowd density.

LBP operator is based on labeling the pixels of an image by thresholding the 3 x 3-neighborhood of each pixel with the center value and considering the result as a binary digit. Then, a binary number is obtained by concatenating all binary values in a clockwise direction, starting from the top left neighbor. Thus, for a given pixel at (x_c, y_c) position,

the LBP code in decimal form is defined as:

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} S(i_p - i_c)2^p \tag{4.1}$$

where i_c and i_p denote, respectively, the gray values of the center pixel and the P surrounding pixels. S refers to a thresholding function defined as: $S(x) = \begin{cases} 1 & \text{if } (x \geq 0) \\ 0 & \text{otherwise} \end{cases}$

In our proposed approach, each image patch is spatially divided into several non-overlapping blocks from which LBP codes are computed. Block-based LBP is used to better preserve local information. Then, histogram of each block is extracted by collecting the occurrence of LBP codes. Finally, the histogram pieces computed from different blocks are concatenated into a single histogram sequence to represent a given image patch. Assume that each image patch is divided into M blocks $\{B_1, B_1, .., B_M\}$, the histogram of each image patch is formulated as follows:

$$\begin{aligned} H &= ((h_0^1, h_1^1, ..., h_{L-1}^1), ..., (h_0^M, h_1^M, ..., h_{L-1}^M))); \\ h_l^j &= \textstyle\sum_{(x,y)\in B_j} f\{LBP(x,y) = l\} \end{aligned} \tag{4.2}$$

where $[0, ..., L-1]$ denotes the range of gray levels in LBP map, and f is defined as: $f\{A\} = \begin{cases} 1 & \text{if } (A \text{ is true}) \\ 0 & \text{otherwise} \end{cases}$

Given different patch sizes, it is important to apply block normalization to each feature vector (i.e. LBP histogram sequence defined in (4.2)). For this purpose, $L1 - sqrt$ [26] defined as follows is used:

$$H = \sqrt{H/(||\ H\ ||_1 + \varepsilon)} \tag{4.3}$$

where ε is a small constant.
The histogram sequence defined in (4.3) is used as texture descriptor. An overview of the block-based LBP extraction and the histogram normalization on image patch is shown in Figure 4.1. For more accuracy, we resort to dimensionality reduction techniques in order to reduce the dimension of feature vector (L x M) before performing the classification step.

4.4.2 Discriminative subspace learning

As described in the previous section, the LBP feature vector extracted from an image patch is high-dimensional, which brought the inconvenience for the modeling and classification steps due to the so-called “curse of dimensionality". Moreover, the feature vector contains substantial amount of component dimensions which is irrelevant to the underlying crowd density and could have even a negative effect on the classification performance. One simple

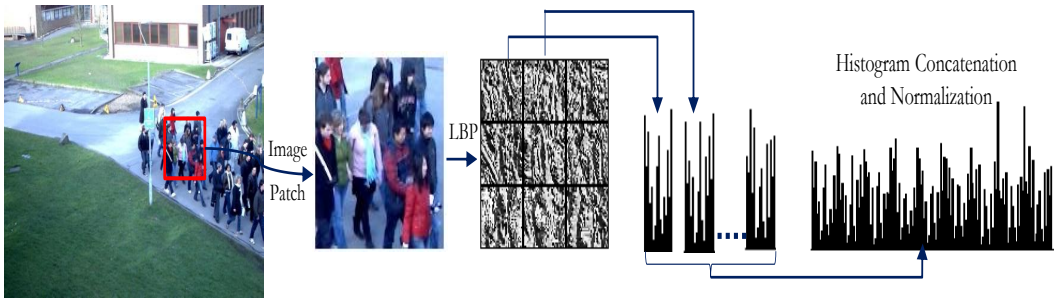


Figure 4.1: *Block-based LBP extraction and Histogram sequence normalization*

way to handle this problem is to apply the so called *uniform patterns* [88]. But, the use of uniformity measure has the limitation of loosing some texture information, which is not important for crowd measurement. That is why, we instead propose to use dimensionality reduction techniques to alleviate the effect of high-dimensional feature vector.

Linear Discriminant Analysis (LDA) is a well-known, simple, but efficient approach to dimensionality reduction, and is widely used in various classification problems. It aims at finding an optimized projection W_{opt} that projects D dimensional data vectors U into a d dimensional space by: $V = W_{opt}U$, in which intra-class scatter (S_W) is minimized while the inter-class scatter (S_B) is maximized. S_W and S_B are determined according to:

$$S_W = \sum_{j=1}^{c} \sum_{i=1}^{l_j} (u_i^j - \mu_j)(u_i^j - \mu_j)^T, \tag{4.4}$$

and

$$S_B = \sum_{j=1}^{c} N_j (\mu_j - \mu)(\mu_j - \mu)^T, \tag{4.5}$$

where u_i^j is the i^{th} sample of of class j, μ_j is the mean of class j, c is the number of classes, and N_j is the number of samples in class j. W_{opt} is obtained according to the objective function:

$$W_{opt} = \arg\max_W \frac{W^T S_B W}{W^T S_W W} = [w_1, \ldots, w_g] \tag{4.6}$$

where $\{w_i | i = 1, \ldots, g\}$ are the eigenvectors of S_B and S_W which correspond to the g largest generalized eigenvalues according to:

$$S_B w_i = \lambda_i S_W w_i, i = 1, \ldots, g \tag{4.7}$$

Note that there are at most $c - 1$ non-zero generalized eigenvalues, so g is upper-bounded by $c - 1$. Since $S_W$is often singular, it is common to first apply Principal Component Analysis (PCA) [61] to reduce the dimension of the original vector. This dimensionality

reduction process of PCA followed by LDA is well accepted in face recognition domain and is commonly referred to as “Fisherface" [9]. In our work, we adopt the same strategy for crowd density estimation problem.

4.5 Multi-Class SVM classifier

Once the dimensionality reduction techniques (which stand to PCA+LDA) are applied on block-based LBP, the resulting feature vectors are classified into different crowd levels by applying Support Vector Machine (SVM) [24].

4.5.1 Baseline multi-class SVM method

Since crowd density estimation involves multiclass classification and SVM is originally two-class based pattern classification algorithm, the problem is addressed by combining several binary SVM classifiers. The most frequently used techniques are: one-vs-rest, and one-vs-one, where for a k-class problem k, and $k(k-1)/2$ binary SVM classifiers are performed, respectively. According to [56], one-vs-one usually perform better than one-vs-rest.

Let consider a training set of N pairs $(v_1, l_1), ..., (v_N, l_N)$, where $v_i \in \mathbb{R}^d$ refers to the reduced feature vector of a given image patch i, and $l_i \in \{C_1, ..., C_5\}$ is the label which indicates the crowd density level of a sample v_i. Using one-vs-one [69], to classify an input feature vector v_i, $k(k-1)/2$ binary SVM classifications are performed, in which SVM finds the maximum-margin hyper-plane to separate the data by:

$$class(v_i) = sign(\sum_{j=1}^{N} \alpha_j l_j K(x_j, v_i) + b) \tag{4.8}$$

where $\{x_j, j \in [1, ..., N]\}$ are the support vectors. For each binary classification, samples from two classes are trained, the classifier assigns the instance to one of the two classes and consequently the vote of that class is increased by one. Then, for the final decision of the crowd level, the output of all the decision functions of binary SVMs are combined. For this purpose, “Max Wins" voting strategy is employed, in which the class of a given feature vector is the one that gets the highest number of votes.
In our experiments, two types of SVM kernels are evaluated:
Linear kernel: $K(x, y) = x.y$
Radial Basis Function (RBF) kernel: $K(x, y) = e^{|x-y|^2/2\sigma^2}$.
An illustration of one-vs-one for crowd level classification can be depicted at Figure 4.2.

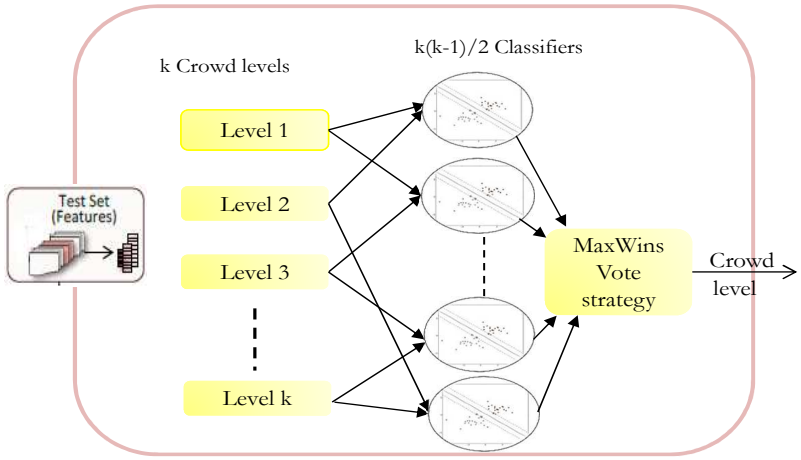


Figure 4.2: *One-vs-one multi-classification for crowd density estimation problem*

4.5.2 MultiClass SVM based on Graded Relevance Degrees

At this stage, we intend to improve the classification accuracy while maintaining less computational cost over the existing multiclass SVM approaches. Our proposed algorithm consists of combining $(k-1)$ binary classifiers into a multiclass classifier. It proceeds as it is shown in Algorithm 2.

The main idea is to reassess each binary SVM classifier using relevance scores. In other words, we go beyond a binary crowd subdivision by assigning different crowd levels to the classified samples. This automatic graded crowd judgments is performed using fuzzy membership score which was proposed in [97] as a measure to quickly build graded ground truths in binary labeled databases without involving manual effort.
Since a binary SVM classification aims at finding a hyperplane that optimally separates two classes in the feature space, the distance from the hyperplane can be used to measure how much a sample is representative in one class. Therefore, the decision value $f(v_s)$ of each training sample v_s is calculated, then a fuzzy score [97] is defined as the positive/negative class posterior probability: $\sigma_s = p(l_s = sign(f(v_s))|f(v_s))$ with a parametric model based on fitting a sigmoid function:

$$\sigma_s = \frac{1}{1+\exp(af(v_s)+b)} \tag{4.9}$$

where a and b parameters are adapted on the training step.
According to the fuzzy relevance scores, the positive and negatives training samples of each classifier are sorted and different thresholds are defined so that, we can re-categorize the samples in each set into different graded crowd levels.

Our proposed SVM multi-classification algorithm can be applied for any multiclass

Algorithm 2: MultiClass SVM

Input: Training set $(v_1, l_1), ..., (v_N, l_N)$
Output: Multiclass Classifier
Training: Binary SVMs and graded relevance scores
for $j = 1$ **to** $(k-1)$ **do**

- For all samples from C_1 to C_j classes, set labels to (-1) and all samples from C_{j+1} to C_k, set labels to (+1)
- Train j^{th} binary SVM
- Classify the training samples
- if $(j > 1)$, compute fuzzy scores σ_n for all training samples v_n classified as (-1) and define $(j-1)$ thresholds by splitting the curve of sorted relevance scores into equally spaced intervals.
- if $(j < (k-1))$, compute fuzzy scores σ_p for all training samples v_p classified as (+1) and define $(k-j-1)$ thresholds by splitting the curve of sorted relevance scores into equally spaced intervals.

end for
Testing:Classification of a new sample z_l
for $j = 1$ **to** $(k-1)$ **do**

- Classify z_l by j^{th} model
- **if** (z_l is classified as (+1))
  if $(j = 1)$ $class^j(z_l) \leftarrow C_1$ else use the defined thresholds to decide $class^j(z_l)$
  else
  if $(j = k-1)$ $class^j(z_l) \leftarrow C_k$ else use the defined thresholds to decide $class^j(z_l)$
  end if

end for
The class getting the highest votes determines the instance class, if the same number of votes, the decision is made based on the relevance scores.

problem, where classes are related by monotonically increasing relevance degrees. Furthermore, this algorithm incurs at least two advantages: First, the computation time is decreased because only $(k-1)$ binary SVMs are performed. Second, each binary classification can be converted to multiclass classification using relevance scores σ_s. An example of the first split (first binary SVM) that labels samples from C_1 (free flow) as (-1) and samples from C_2 to C_5 (from restricted to jammed flow) as (+1) is illustrated in Figure 4.3.

4.6 Experimental Results

4.6.1 Dataset

The proposed approach is evaluated within PETS 2009 public dataset [41]. In particular, we select some frames from S_1 and S_2 Sections. Then, we define the different crowd levels

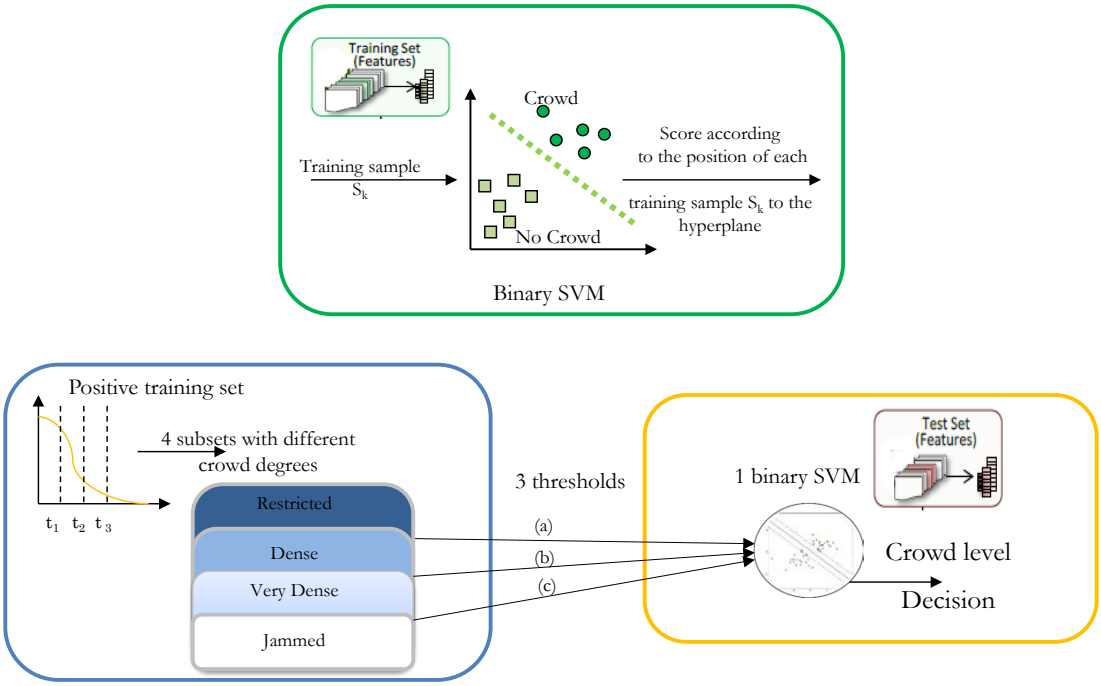


Figure 4.3: *Proposed Multi-SVM based on relevance degrees*

[94] according to the range of people in $13m^2$, see Table 4.1.
Actually this area ($13m^2$) corresponds to the real size of an image block of size 226 x 226 (in the bottom of a frame). Then, the remaining image patches from bottom to top are carefully selected with different patch sizes according to their spatial localization in order to attenuate the effects of perspective distortions before estimating crowd levels. The extraction of multi-scale patches is shown in Figure 4.4.

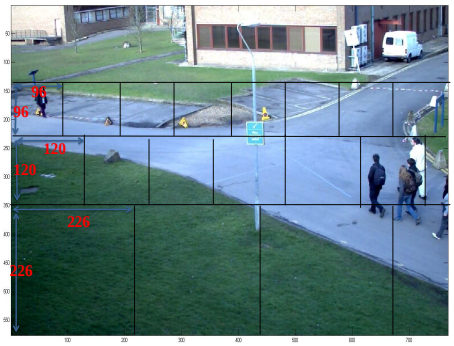


Figure 4.4: *Multi-scale patches*

Afterwards, we manually label these image patches according to the congesting degrees

Levels of Crowd Density	Range of Density ($people/m^2$)	Range of People
Free Flow	< 0.5	< 7
Restricted Flow	0.5-0.8	7-10
Dense Flow	0.81-1.26	11-16
Very Dense Flow	1.27-2.0	17-26
Jammed Flow	> 2.0	> 26

Table 4.1: Definition of different crowd levels according to the range of density, and according to the range of people in an area of an approximate size $13m^2$.

of the crowd defined in Table 4.1. Using PETS dataset, we could not reach level 5 of the crowd (jammed flow), therefore, only four levels are experimented. For each crowd level, 200 image patches are selected, 100 for training and another 100 patches for testing, totally 800 images patches for all the crowd levels. This results in a 4-class training set and a testing set of 400 samples each.
SVM parameters are optimized within the training set, using cross-validation (we randomly choose 20 patches to tests, for each crowd level). The same strategy is adopted for selecting PCA and KNN parameters.

4.6.2 Experiments

As described in Section 4.4, LBP features are extracted from 3 x 3 blocks in each patch sample, and PCA and LDA subspaces are trained with the labeled training set. The projections of training samples are further used for training multi-class SVM classifiers as described in Section 4.5. The performance is evaluated in two steps. First, for each test sample, the feature vector using block-based LBP is projected into the learned PCA and LDA subspaces, and is identified as one of the four classes by the multi-class SVM classifiers following one-vs-one strategy. The top-1 identification accuracy is reported. One-vs-one is chosen for evaluating the performance of texture features, because it has been demonstrated in the literature that it gives better results compared to other multiclass methods [56]. Second, the Receiver Operating Characteristics (ROC) curve of each class is reported to demonstrate the discriminative power of our proposed feature for each crowd density level separately. Furthermore, in our experiments, both of linear and RBF SVM kernels are evaluated. Their performances are compared to K-Nearest Neighbor (KNN) classifier. We also compare our proposed feature to other texture features, namely, HOG [26], Gabor wavelet [116] and GLCM [83]. Finally, we compare the performance of our proposed multiclass SVM algorithm based on relevance degrees to one-vs-one and one-vs-rest methods.

4.6.3 Results and analysis

Since a key step in crowd density estimation is the choice of texture feature, we compare our proposed feature LBP+DR (which stands for LBP+PCA+LDA) with other frequently used texture features: HOG, Gabor, and GLCM. In addition, the performance of our proposed feature is compared to the classification accuracy achieved using SVM on the raw LBP features, see Figure 4.5.

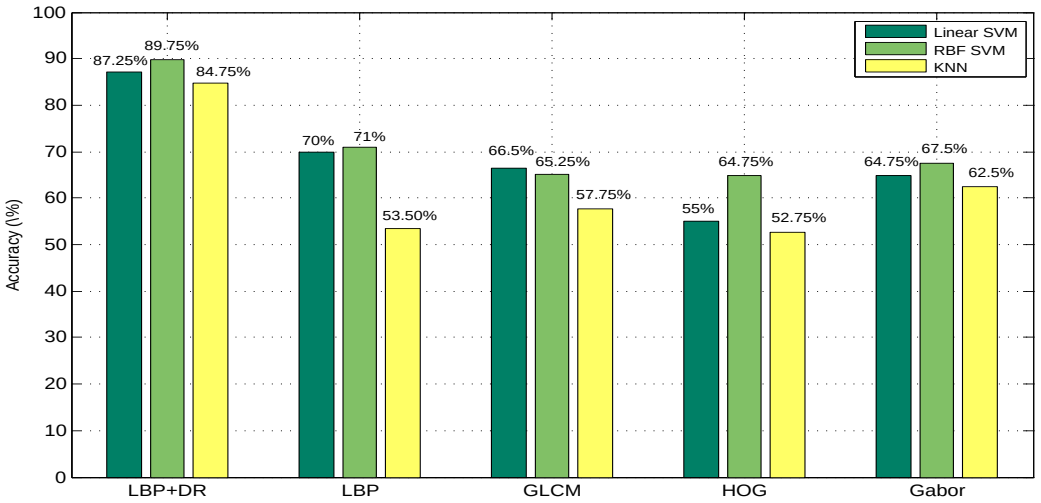


Figure 4.5: *Comparisons of our proposed feature (LBP+DR) to other texture features LBP, GLCM, HOG, and Gabor using one-vs-one SVM (for both Linear and RBF kernels) and KNN classifiers*

In this Figure, we also include a comparison between SVM (for both linear and RBF kernels) and KNN classifiers. As shown in this Figure, the comparison of our proposed feature to LBP features demonstrates the substantial improvement made by the dimensionality reduction on LBP features in the classification accuracy. The classification accuracy is improved by around 20% using RBF kernel (and around 16% using linear kernel), after applying dimensionality reduction techniques over directly using raw LBP features. These results demonstrate the relevance of the discriminant feature selection process.
Besides, these comparisons clearly show that our proposed feature (LBP+DR) outperforms all the other texture features. In addition, the classification accuracy using SVM shows substantial improvement over KNN classification (with better results of RBF kernel compared to linear kernel). In overall, the combination LBP+DR+SVM (using RBF kernel) gives the best results in terms of classification accuracy (89.75%) with a significant margin compared to the other tested texture features. As illustrated in Figure 4.5, SVM classifier using RBF kernel has almost the best overall performance for all aforementioned texture features, and is thus selected for next experiments.

At this stage, we intend to evaluate the accuracy of texture features for each crowd

level independently from the others; it means to explore how much each texture feature is discriminative to a specific level. To achieve this goal, ROC curve for each crowd level class is reported, see Figure 4.6.

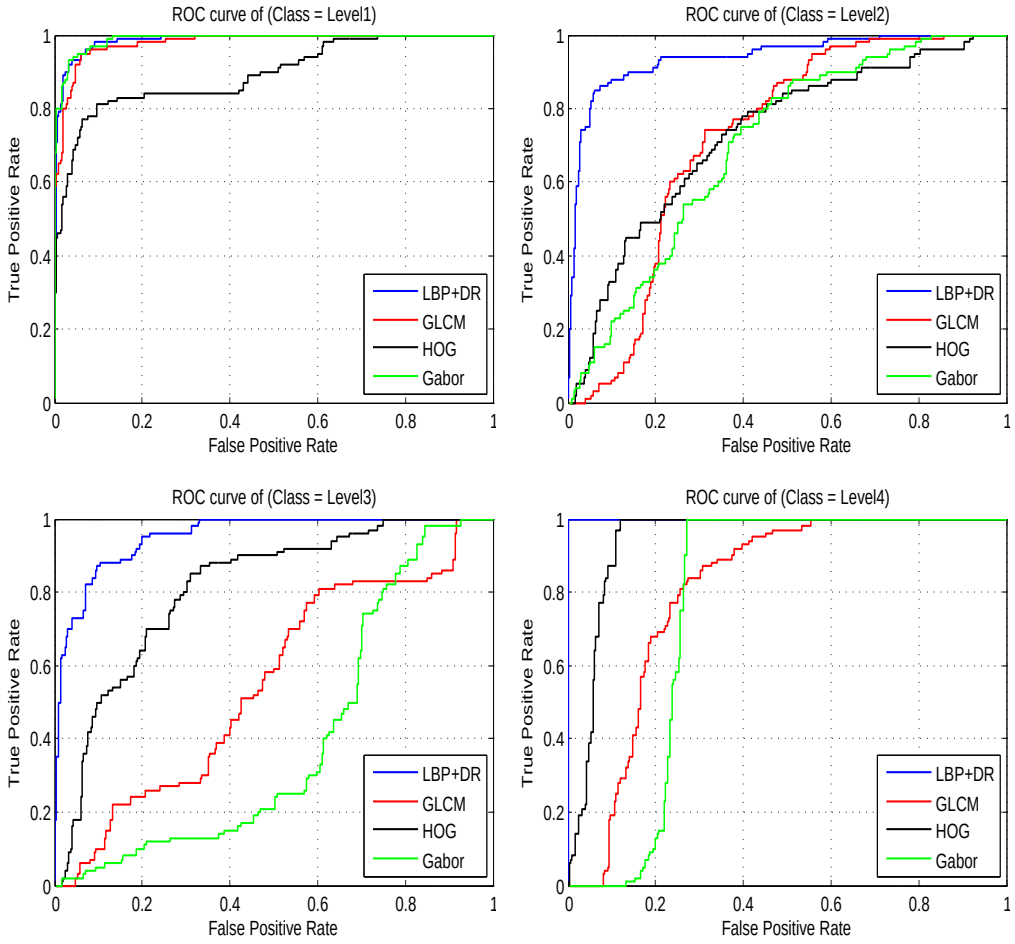


Figure 4.6: *Comparisons of the ROC curves of the proposed feature (LBP+DR) with other texture features (GLCM, HOG, Gabor) for 4 different crowd levels (free, restricted, dense, very dense flows) using RBF kernel for SVM classification*

Then, the performance of each texture feature in a specific crowd level is measured by computing the area under the curve (AUC) and the accuracy (ACC), the results are reported in Table 4.2.

As it is shown in Figure 4.6 and also demonstrated in Table 4.2, LBP+DR outperforms all other texture features at any crowd level. Also, the results show that the tested texture features presented better discriminative ability for free and very dense flows (level 1 and level 4) compared to restricted and dense flows (level 2 and level 3). So, most of the confusions in the classification step are in the intermediate classes, however, the results

Features	Level 1		Level 2		Level 3		Level 4	
	AUC	ACC(%)	AUC	ACC(%)	AUC	ACC(%)	AUC	ACC(%)
LBP+DR	0.98	95.25	0.93	91.50	0.95	87.00	1.00	97.50
GLCM	0.98	91.75	0.72	65.75	0.55	75.00	0.80	39.50
HOG	0.89	87.75	0.72	76.00	0.80	76.25	0.94	90.25
Gabor	0.99	91.75	0.70	70.75	0.39	65.75	0.76	79.25

Table 4.2: Evaluation of texture features for each crowd level in terms of AUC and ACC

show that LBP+DR succeeds to overcome this difficulty, in terms of AUC and ACC.

Finally, we intend to evaluate the performance of our proposed multiclass SVM algorithm based on relevance scores. To achieve this goal, the performance of LBP+DR feature using our algorithm is compared to one-vs-one and one-vs-rest methods using linear and RBF kernels. The results are reported in Table 4.3.

Multiclass method	Linear SVM	RBF SVM	Number of binary SVM
One-vs-one	87.25%	89.75%	6
One-vs-rest	72.25%	84.00%	4
Proposed algorithm	88.25%	89.00%	3

Table 4.3: Comparisons of our proposed multiclass SVM algorithm to one-vs-one and one-vs-rest algorithms for both linear and RBF kernels using LBP+DR features

In Table 4.3, the classification accuracy using our proposed multiclass SVM is reported and compared to one-vs-one and one-vs-rest. We also include a comparison between these methods in terms of number of binary SVMs. According to the results, the proposed algorithm has less computational cost compared to the other multiclass SVM techniques. And its evaluation in terms of accuracy shows substantial improvement over one-vs-rest while maintaining comparable accuracy compared to one-vs-one.

4.7 Conclusion

In this Chapter, we proposed a novel approach for crowd density estimation at patch level. It consists of learning a discriminant subspace of the high-dimensional LBP raw feature vector where samples of different crowd density are optimally separated. In addition, an alternative algorithm for multiclass SVM based on relevance scores is proposed. The ef-

fectiveness of the proposed approach is evaluated on PETS dataset, and the results demonstrate the effects of low-dimensional compact representation of LBP on the classification accuracy. The performance of the proposed framework is also compared to other frequently used features in crowd density estimation. Our proposed algorithm outperforms other methods with a significant margin. Also, the performance of the proposed multiclass SVM algorithm is compared to other frequently used algorithms for multi-classification problem and the proposed algorithm gives good results while reducing the complexity of the classification problem.

CHAPTER 5

Crowd Density Map Estimation Using Sparse Feature Tracking

5.1 Introduction

In this chapter, we propose a novel approach for crowd density measure, in which local information at pixel level substitutes a global crowd level or a number of people per-frame. The proposed approach consists of generating fully automatic crowd density maps using local features as an observation of a probabilistic crowd function. It also involves a feature tracking step which allows excluding feature points belonging to the background. This process is favorable for the later density function estimation since the influence of features irrelevant to the underlying crowd density is removed. Furthermore, we propose an evaluation methodology of the crowd density maps which is based on estimating a linear transformation that minimizes the mismatches between the feature representation and the ground truth density.

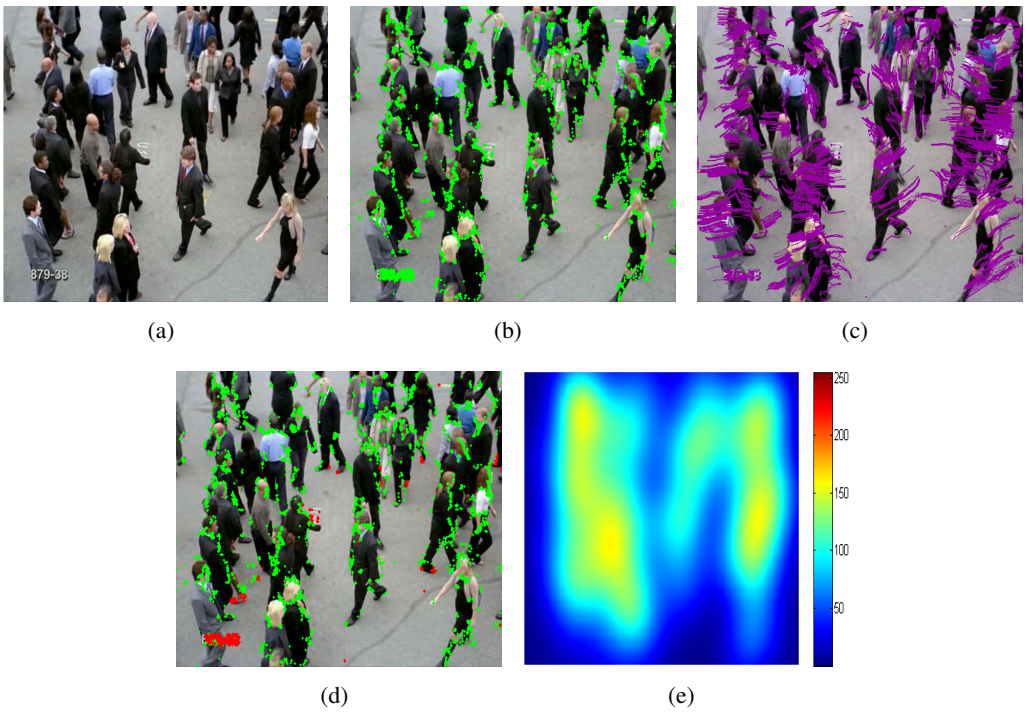
5.2 Motivation

Crowd density analysis has been studied as a major component for crowd monitoring and management in visual surveillance systems, its estimation is fundamental to detect potential overcrowd. As mentioned in the previous chapters, in the simplest forms, the used crowd density measures could be the number of persons or alternatively the level of the crowd. These forms of crowd density analysis have the limitation of giving a global information for the entire image and discarding local information about the crowd. From this perspective, we resort to crowd information at local level by computing crowd density maps. This alternative solution is indeed more appropriate as it enables both the detection and the location of potentially crowded areas.

The proposed crowd density map is typically based on using local features as an observation of a probabilistic crowd function. Also, a feature tracking step is involved in the crowd density process. In fact, considering all extracted local features brings an inconvenience to the density function estimation as a substantial amount of components are irrelevant to the underlying crowd density. Therefore, we propose using motion information to alleviate this effect.

The remainder of the Chapter is organized as follows: In the next Section 5.3, we present our proposed approach for crowd density map estimation. An evaluation methodology of the proposed density map is introduced in Section 5.4. Detailed experimental results follow in Section 5.5.

5.3 Crowd Density Map Estimation



(a) (b) (c)

(d) (e)

Figure 5.1: Illustration of the proposed crowd density map estimation using local features tracking: (a) Exemplary frame, (b) FAST Local features (c) Feature tracks (d) Distinction between moving (green) and static (red) features - red features at the lower left corner are due to text overlay in the video (e) Estimated crowd density map (the color map Jet is used so red values represent higher density where blue values represent low density)

Since generating locally accurate crowd density maps is more helpful than computing only an overall density or a number of people in a whole frame, we propose substituting global information per-frame by local information at pixel level. Our proposed approach proceeds as follows: First, local features are extracted to infer the contents of each frame under analysis. Then, we perform local features tracking using the Robust Local Optical Flow algorithm from [111] and a point rejection step using forward-backward projection. To accurately represent the motion within the video, the estimation of optical flow be-

tween consecutive frames is extended to build trajectories. The generated feature tracks are thereby used to remove static features. Finally, crowd density maps are estimated using Gaussian symmetric kernel function.
An illustration of the density map modules is shown in Figure 5.1. The remainder of this section describes each of these system components.

5.3.1 Extraction of local features

One of the key aspects of crowd density measurements is crowd feature extraction. Under the assumption that regions of low density crowd tend to present less dense local features compared to a high-density crowd, we propose to use local feature points as a description of the crowd by relating dense or sparse local features to the crowd size. For this purpose, we first extract local features, then, the crowd density map is estimated by measuring how close local features are.

For local features, we assess Features from Accelerated Segment Test (FAST) [104], Scale-Invariant Feature Transform (SIFT) [76], and Good Features to Track (GFT) [117]. The reason behind selecting these features for crowd measurement is as follows: FAST was proposed for corner detection in reliable way. It has the advantage of being able to find small regions which are outstandingly different from their surrounding pixels. Besides, FAST was used in [13] to detect dense crowds from aerial images and the derived results demonstrate a reliable detection of crowded regions. SIFT is another well-known texture descriptor, that defines interest point locations are defined as maximum/minimum of the difference of Gaussians in scale-space. Under this respect, SIFT is rather independent of the perceived scale of the considered object which is appropriate for crowd measurements. These two aforementioned features are compared to the classic feature detector GFT, which is based on the detection of corners containing high frequency information in two dimensions and typically persist in an image despite object variations.

The extracted features will be further used as observations of the probability density function. But since the probability density function has to correspond to the density of crowds, a feature selection process is required to remove features which are not relevant to the crowd density.

5.3.2 Local features tracking

Using the extracted features directly to estimate the crowd density map without a feature selection process might incur at least two problems: First, the high number of local features increases the computation time of the crowd density. As a second and more important effect, the local features contain components from the background which are irrelevant to the crowd density. Thus, we need to add a separation step between foreground and background entities to our framework. This is done by assigning motion information to the detected local features in order to distinguish between moving and static ones. Based on

the assumption that only persons are moving in the scene, these can then be differentiated from background by their non-zero motion vectors.

Motion estimation is performed using the Robust Local Optical Flow (RLOF) [111], [110], which computes accurate sparse motion fields by means of a robust norm[1]. The motion vector $\mathbf{d}$ is computed by a minimization of the shrinked Hampel norm with the parameters σ_1, σ_2 defining the treatment of outliers:

$$\rho(y, \boldsymbol{\sigma}) = \begin{cases} y^2 & , |y| \leq \sigma_1 \\ \sigma_1 \sigma_2 & , |y| \geq \sigma_2 \\ \frac{\sigma_1}{\sigma_1 - \sigma_2}(|y| - \sigma_2)^2 + \sigma_1 \sigma_2 & , \text{else} \end{cases} . \tag{5.1}$$

More details about RLOF algorithm can be found in [111], [110]. A common problem in local optical flow estimation is the choice of feature points to be tracked. Depending on texture and local gradient information, these points often do not lie on the center of an object but rather at its borders and can thus be easily affected by other motion patterns or by occlusion. While RLOF handles these noise effects better than the standard Kanade-Lucas-Tomasi (KLT) feature tracker [123], it still is not prone against all errors. This is why, we establish a forward-backward verification scheme where the resulting position of a point is used as input to the same motion estimation step from the second frame into the first one. Points for which this “reverse motion" does not result in their respective initial position are discarded. For all other points, motion information is aggregated to form longterm trajectories by connecting motion vectors computed on consecutive frames. This results a set of p_k trajectories in every time step k:

$$\begin{aligned} \mathcal{T}_k &= \{T_1^k, ..., T_{p_k}^k| = \\ T_i^k &= \{X_i(k - \Delta t_i^k), Y_i(k - \Delta t_i^k), ..., X_i(k), Y_i(k)\}\} \end{aligned} \tag{5.2}$$

where Δt_i^k denotes temporal interval between the start and the current frames of a trajectory T_i^k. $(X_i(k - \Delta t_i^k), Y_i(k - \Delta t_i^k))$, and $(X_i(k), Y_i(k))$ are the coordinates of the feature point in its start and current frames respectively. The advantage of using trajectories in our system instead of computing the motion vectors only between two consecutive frames is that outliers are filtered out and the overall motion information is more reliable and less affected by noise.

5.3.3 Kernel density estimation

After generating trajectories, the following goal is to remove static features. These are identified by comparing displacements of the generated trajectories to a small constant ζ. It means to compare the overall mean motion Γ_i^k of a trajectory T_i^k is to a certain

[1] www.nue.tu-berlin.de/menue/forschung/projekte/rlof

threshold ζ. Moving features are then identified by the relation $\Gamma_i^k > \zeta$ while the others are considered as part of the static background. As a result, the separation between foreground and background entities is improved and the number and position of the tracked features undergo an implicit temporal filtering step which makes them smoother.

After filtering out static features, the crowd density map is defined as a kernel density estimate based on the positions of local features. Starting from the assumption of a similar distribution of feature points on the objects, the observation can be made that the more local features come towards each other, the higher crowd density is perceived. For this purpose, a probability density function (pdf) is estimated using a Gaussian kernel density. For a given video sequence of N frames $\{I_1, I_2, ..., I_N\}$, if we consider a set of m_k local features extracted from a frame I_k at their respective locations $\{(x_i, y_i), 1 \leq i \leq m_k\}$, the corresponding density map C_k is defined as follows:

$$C_k(x,y) = \frac{1}{\sqrt{2\pi}\sigma} \sum_{i=1}^{m_k} \exp -(\frac{(x-x_i)^2 + (y-y_i)^2}{2\sigma^2}) \tag{5.3}$$

where σ is the bandwidth of the 2D Gaussian kernel. The resulting density function defines our proposed crowd density map which gives valuable information about the local distribution of people in the scene.

To take into the effects of perspective distortions, one way to do that is to make σ space variant according to the perspective map. Since the perspective distortions are usually handled manually, we preferred to keep our proposed approach fully automatic and we rather rely on the scale invariant aspect of local features. Thus, in our approach σ is a constant.

5.4 Evaluation methodology

After generating crowd density maps using sparse feature tracks, we aim at evaluating these maps. The following methodology is adapted: we consider that an accurate estimation of the density map could adequately represent the spatial distribution of people in the scene. For this purpose, we define a ground truth density function as a kernel density estimate based on annotated person detections. And, we assume that an optimal feature representation can be produced by simple linear weighting of the ground truth density. Hence, for an input frame I_k from a video sequence V, given a set of annotated detections $\phi_k = \{\varphi_1^k, ..., \varphi_{l_k}^k\}$, $\varphi_i^k = \{xc_i^k, yc_i^k, h_i^k, w_i^k\}$, where (xc_i^k, yc_i^k), h_i^k, w_i^k denote, respectively, the center coordinates, the height, and the width of the annotated bounding box φ_i^k. The corresponding ground truth density G_k is defined as:

$$G_k(x,y) = \sum_{i=1}^{l_k} \frac{1}{\sqrt{2\pi}\sigma_i^k} \exp -(\frac{(x-xc_i^k)^2 + (y-yc_i^k)^2}{2{\sigma_i^k}^2}) \tag{5.4}$$

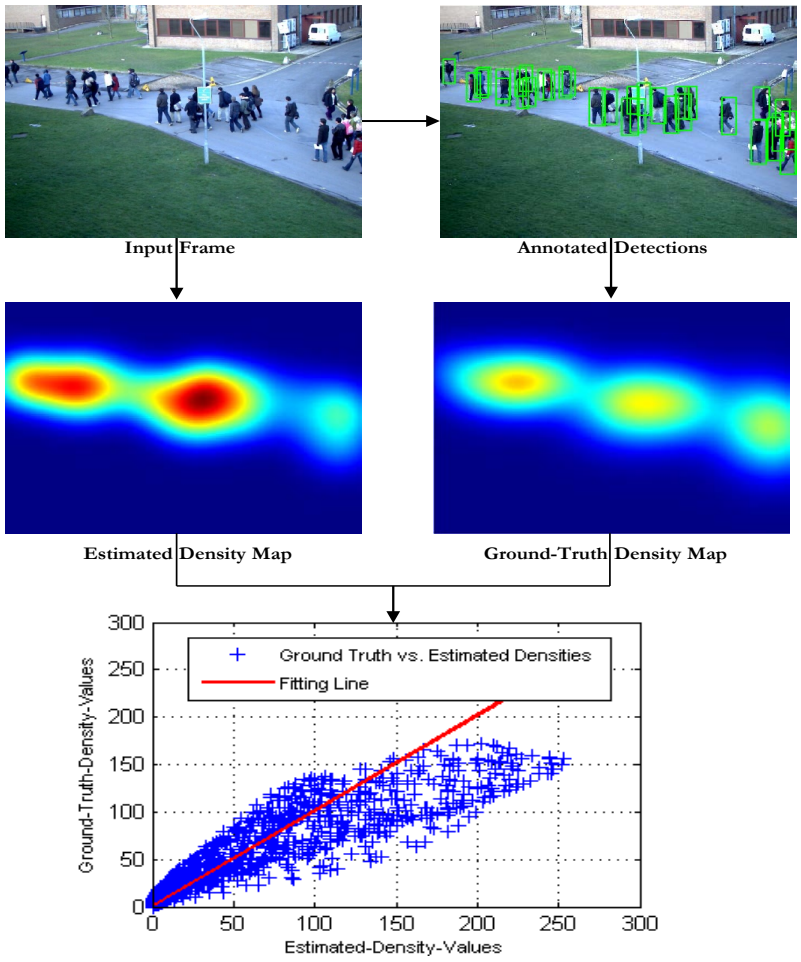


Figure 5.2: Flowchart of the evaluation methodology of crowd density map: The ground truth density is estimated using annotated person detection. These ground truth values are plotted vs. the estimated density values to approximate the linear transformation mapping the estimated to the ground truth values. The distortions from the fitting line are used for the evaluation.

where σ_i^k corresponds to the size of the bounding box φ_i^k, i.e. $\sigma_i^k = h_i^k.w_i^k$.

At this stage, our objective is to find a way to automatically evaluate the estimated crowd density map. The idea is inspired from [71], where the goal is to learn a linear transformation that minimizes the error between a feature representation and the ground truth from a set training samples. However, in our work, we intend to approximate this linear transformation rather from the testing samples.

Given the estimated density maps $\{C_i\}_{i=1}^N$ and their corresponding ground truth density maps $\{G_i\}_{i=1}^N$ of a video sequence V, we aim at estimating the linear transformation mapping C_i to G_i, $i \in \{1...N\}$ with the least mismatches between them. Similar to [71],

the parameter vector Ω of this linear transformation is defined as:

$$\begin{gathered}\Omega = \underset{\omega}{\operatorname{argmin}}(\omega^T\omega + \lambda \sum_{i=1}^{N} Dist(G_i(.), C_i'(.|\omega)), \\ C_i'(.|w) = w^T C_i(.)\end{gathered} \tag{5.5}$$

where λ is a scalar hyperparameter controlling the regularization strength while $Dist$ is the distance measuring the loss i.e. the mismatch between the estimated and the ground truth densities. $Dist$ is chosen in [71] to be the regularized MESA distance since their goal is an overall count. This choice does not match our goal of evaluating the local distribution of density values. Thus, more appropriate choice of $Dist$ could be an L_p metric, which turns (5.5) to a typical linear regression problem, where each sample corresponds to a pixel rather than the whole image. The distortions from the fitting regression line could be used to find the mismatches between the ground truth and the estimated density values, see Figure 5.2.

5.5 Experimental Results

5.5.1 Datasets and Experiments

The proposed approach for crowd density map estimation is evaluated within challenging crowd scenes from multiple video datasets. In particular, we select some videos from PETS 2009 [41], UCF [4], and data driven crowd analysis [103] public datasets. As described in Section 5.3, local features are extracted and tracked in each frame under analysis. The moving local features are further used for estimating the crowd density map. The effectiveness of our proposed approach is demonstrated in two steps. First, we compare FAST to other local features, namely, SIFT [76], and GFT [117]. Furthermore, we compare the results using feature tracks to the results using foreground segmentation [138] to demonstrate the advantages of building trajectories in our system.

For evaluation, we adapt the methodology described in Section 5.4. Once the linear transformation is applied, the evaluation is performed by comparing the projected estimated densities to the ground truth densities. Two quality metrics are used to compute error statics with respect to the ground truth data:

- MAE (mean-absolute-error) between the ground truth densities G_k and the estimated densities C_k' after applying linear transformation:

$$E = \frac{1}{P} \sum_{(x,y)} |C_k'(x,y) - G_k(x,y)| \tag{5.6}$$

  where P is the total number of pixels.

- Percentage of bad density pixels:

$$B = \frac{1}{P} \sum_{(x,y)} (|C'_k(x,y) - G_k(x,y)| > \tau_d) \tag{5.7}$$

where τ_d is a density error tolerance.

In addition to these quality metrics computed over the whole image, more evaluations are conducted to assess the discriminative power of the local features to the crowd. Specifically, we split the image regions to Crowd / No Crowd regions using the reference image and the ground truth density map. This consists of the following binary segmentation: if the ground truth density value is below a given threshold, the pixel belongs to not crowded regions $\overline{\mathcal{C}}$, otherwise it belongs to crowded regions $\mathcal{C}$. As a result, the two metrics described above are additionally computed for each of the two regions. For experiments, we use the evaluation metrics listed in Table 5.1.

Symb.	Name	Description
E	$mae - error - all$	MAE density error
$E_{\mathcal{C}}$	$mae - error - crowd$	MAE density error in crowd
$E_{\overline{\mathcal{C}}}$	$mae - error - noncrowd$	MAE density error in no crowd
B	$bad - pixels - all$	bad pixel percentage
$B_{\mathcal{C}}$	$bad - pixels - crowd$	bad pixel percentage in crowd
$B_{\overline{\mathcal{C}}}$	$bad - pixels - noncrowd$	bad pixel percentage in no crowd

Table 5.1: Quality metrics used to evaluate crowd density map with respect to the ground truth data

5.5.2 Results and Analysis

We first report the results of our proposed approach in terms of mean-absolute error in Table 5.2. In this Table, the normalized MAE to the range of data is used in order to insure scale independence and the three evaluations metrics (E, $E_{\mathcal{C}}$ and $E_{\overline{\mathcal{C}}}$) are computed. Also, the results using B, $B_{\mathcal{C}}$ and $B_{\overline{\mathcal{C}}}$ quality metrics are shown in Figure 5.3, where the x-axis corresponds to the density error tolerance (i.e. τ_d defined in (5.7) which varies from zero to 255).

In both Table 5.2 and Figure 5.3, a comparison between the three local features FAST, SIFT, and GFT is shown. Also, the results of a GMM-based crowd density-estimation (which consists of substituting features tracking step by foreground segmentation in crowd density estimation process) are given. These comparisons clearly show that the feature tracking step achieves substantial improvement over using foreground segmentation. That highlights the advantage of using trajectories in our system instead of computing the mo-

tion vectors only between two consecutive frames or by foreground segmentation. Our estimate is more robust to noise and the overall motion information is more accurate. As a result, the number and position of the tracked features undergo an implicit temporal filtering step which improves consistency compared to the separation between foreground and background entities.

By comparing different local features, the evaluations in terms of mean-absolute-error E (**VAL1** in Table 5.2), and in terms of bad pixels percentage (the first column in Figure 5.3) show that the choice of local features in general has limited impact on the performance if we consider all image regions (i.e Crowd / No Crowd), even if a small improvement of FAST features is noted compared to other features. However, a more significant margin between FAST performance and the two other features is shown in crowded regions (using E_C and B_C quality metrics) that could demonstrate the relevance of FAST for density estimation in crowded scenes.

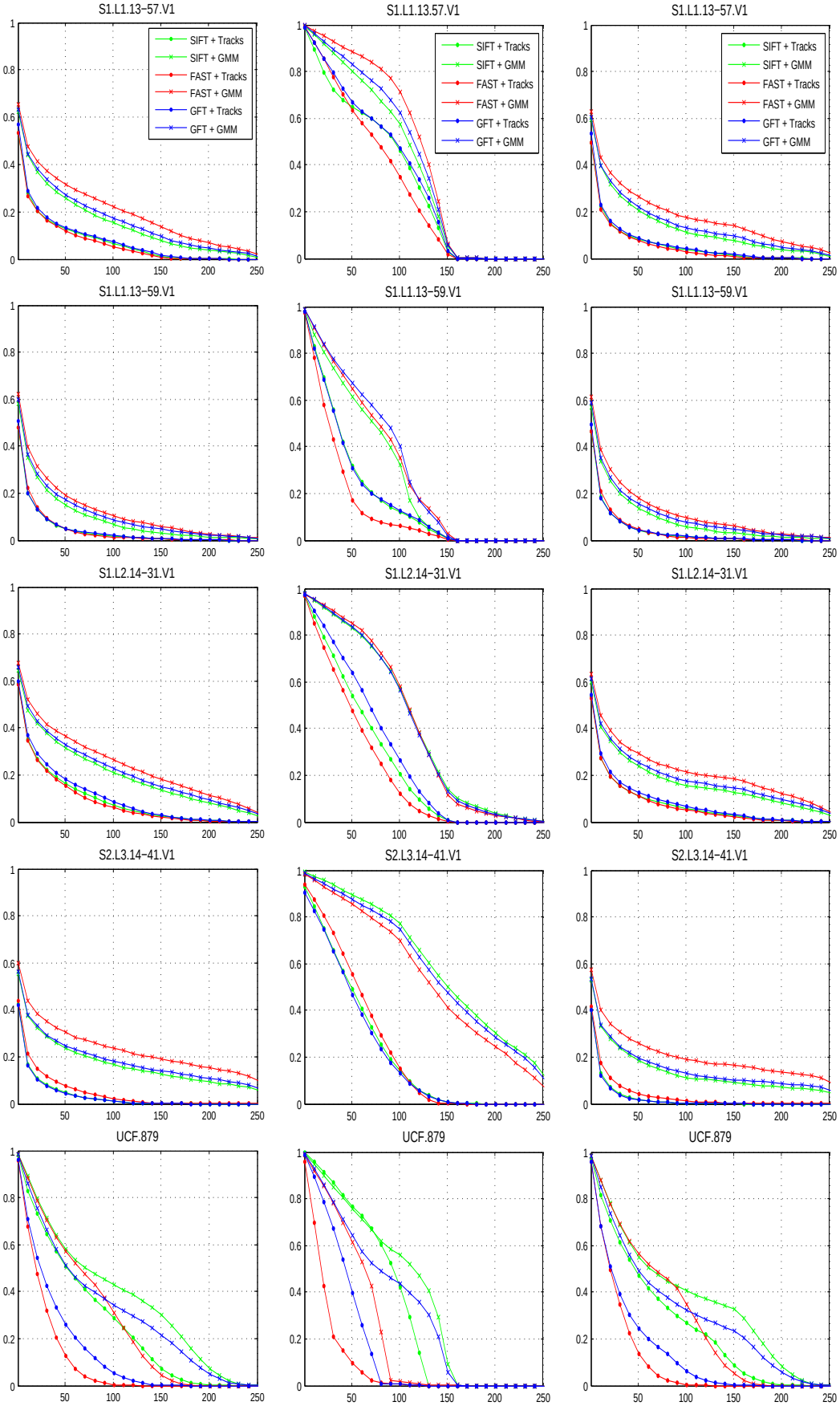
5.6 Conclusion

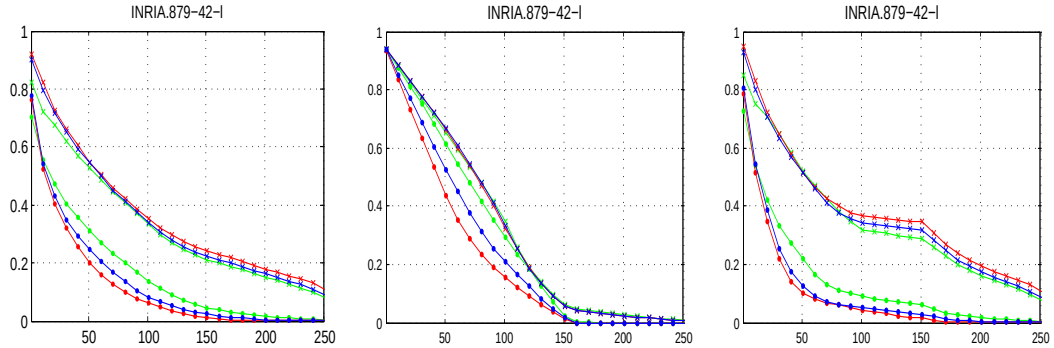
In this Chapter, we present our proposed approach for crowd density estimation which is typically based on using local features as observation of density function. Our approach is extended to feature tracking which enables us to identify objects in the scene that have undergone a sufficient motion to be considered as a person. Consequently, the effort of computation is reduced to the features relevant to the crowd density. In the experimental results, an extensive evaluation on several datasets shows the effectiveness of our approach. Furthermore, we include a comparative study to investigate the discriminative power of different local features to the crowd. These comparisons prove that FAST-based method is robust enough to perform well in both Crowd/No Crowd situations. In addition, the results highlight the relevance of the feature tracking process compared to the foreground segmentation.

The proposed crowd density map characterizes the spatial and temporal variations of the crowd. The spatial variation arises across the frame thanks to the probability density function and temporal variation occurs over the video by the motion information included in the process. Overall, this spatio-temporal crowd information introduced by density maps conveys rich information about the distributions of pedestrians in the scene which could complement other video surveillance applications in failure cases in crowded scenes.

Sequence name	Feature	E	$E_{\overline{\mathcal{C}}}$	$E_{\mathcal{C}}$
PETS S1.L1.13-57	FAST	**0.0670** / 0.2002	**0.0480** / 0.1774	**0.2977** / 0.4368
	SIFT	**0.0729** / 0.1520	**0.0520** / 0.1301	**0.3218** / 0.3844
	GFT	**0.0767** / 0.1661	**0.0553** / 0.1436	**0.3365** / 0.4041
PETS S1.L1.13-59	FAST	**0.0391** / 0.1199	**0.0367** / 0.1147	**0.1342** / 0.2959
	SIFT	**0.0387** / 0.0911	**0.0352** / 0.0857	**0.1796** / 0.2723
	GFT	**0.0398** / 0.1059	**0.0364** / 0.1000	**0.1802** / 0.3059
PETS S1.L2.14-31	FAST	**0.0857** / 0.2428	**0.0682** / 0.2149	**0.2093** / 0.4105
	SIFT	**0.0918** / 0.2018	**0.0715** / 0.1679	**0.2417** / 0.4101
	GFT	**0.1010** / 0.2162	**0.0784** / 0.1845	**0.2736** / 0.4069
PETS S2.L3.14-41	FAST	**0.0443** / 0.2337	**0.0328** / 0.2033	**0.2301** / 0.5422
	SIFT	**0.0320** / 0.1716	**0.0210** / 0.1346	**0.2135** / 0.6015
	GFT	**0.0306** / 0.1836	**0.0202** / 0.1478	**0.2062** / 0.5809
UCF-879	FAST	**0.0997** / 0.2755	**0.1040** / 0.2815	**0.0891** / 0.2253
	SIFT	**0.2601** / 0.3653	**0.2517** / 0.3601	**0.3272** / 0.3844
	GFT	**0.1393** / 0.3118	**0.1359** / 0.3071	**0.1707** / 0.3281
INRIA-879-42_I	FAST	**0.1109** / 0.3599	**0.0876** / 0.3779	**0.2084** / 0.3039
	SIFT	**0.1603** / 0.3277	**0.1320** / 0.3443	**0.2767** / 0.3062
	GFT	**0.1287** / 0.3450	**0.0996** / 0.3582	**0.2381** / 0.3062

Table 5.2: Results of crowd density estimation for three different local feature types (FAST, SIFT, and GFT) and for different test videos in terms of normalized MAE (E, $E_{\mathcal{C}}$ and $E_{\overline{\mathcal{C}}}$). **Val1**/Val2 are the results of our proposed approach using feature tracks, and the results using GMM foreground segmentation


S1.L1.13−57.V1
S1.L1.13.57.V1
S1.L1.13−57.V1
S1.L1.13−59.V1
S1.L1.13−59.V1
S1.L1.13−59.V1
S1.L2.14−31.V1
S1.L2.14−31.V1
S1.L2.14−31.V1
S2.L3.14−41.V1
S2.L3.14−41.V1
S2.L3.14−41.V1
UCF.879
UCF.879
UCF.879
SIFT + Tracks
SIFT + GMM
FAST + Tracks
FAST + GMM
GFT + Tracks
GFT + GMM



a) Quality metric B (b) Quality metric $B_{\mathcal{C}}$ (c) Quality metric $B_{\overline{\mathcal{C}}}$

Figure 5.3: Results of crowd density estimation for different test videos from top to down: PETS S1.L1.13-57 V1, PETS S1.L1.13-9 V1, PETS S1.L2.14-31 V1, PETS S2.L3.14-41 V1, UCF-879, and INRIA.879-42-l. The results are in terms of bad pixels percentage, specifically, quality metric B in column a), $B_{\mathcal{C}}$ in column b), and $B_{\overline{\mathcal{C}}}$ in column c) and the x-axis corresponds to the density error tolerance that varies from 0 to 255. The results are shown for three different local feature types (FAST, SIFT, and GFT), and the proposed approach using feature tracks is also compared to GMM foreground segmentation. This results in 6 curves in each chart.

Part II

Crowd Density-Aware Video Surveillance Applications

CHAPTER 6

Enhancing Human Detection and Tracking in Crowded scenes

6.1 Introduction

Recently significant progress has been made in the field of person detection and tracking. However, crowded scenes remain particularly challenging and can significantly affect the results due to the overlapping detections and occlusions. In this Chapter, we propose to enhance human detection and tracking in crowded scenes using the crowd density map (introduced in previous Chapter). This additional information cue that consists of modeling time-varying dynamics of the crowd density is integrated it into the state-of-the-art detector and the Probability Hypothesis Density (PHD) tracker. In particular, our proposed approach applies a scene-adaptive dynamic parametrization using this crowd density measure. It also includes a self-adaptive learning of the human aspect ratio and perceived height in order to reduce false positive detections. Finally, our improved detection results are extended to tracking in a tracking-by-detection framework.

6.2 Related Works

Automatic detection and tracking of people in video data is a common task in the research area of video analysis and its results lay the foundations of a wide range of applications. In this context, some techniques were proposed to tackle multi-target tracking in crowded scenes. Most of the existing works about tracking in crowded scenes use motion pattern information as priors for tracking. Some of these methods are applied in unstructured crowd scenes [100], while most of them focus on structured scenes [137, 68, 5] where objects do not move randomly, which exhibits clear motion patterns.

In [100], a tracking approach in unstructured environments, where the crowd motion appears to be random in different directions over time, is presented. For this purpose, a topical model, which allows each location in the scene to represent motion in different directions is used. In [137], a Motion Structure Tracker is proposed to solve the problem of tracking in very crowded scenes. In particular, tracking and detection are performed jointly and motion pattern information is integrated in both steps to enforce scene structure constraints. In [68], a probabilistic method exploiting the inherent spatially and temporally

varying structured pattern of crowd motion is employed to track individuals in extremely crowded scenes. The spatial and temporal variations of the crowd motion are captured by training a collection of Hidden Markov Models on the motion patterns within the scene. Using these models, pedestrian movement at each space-time location in a video can be predicted. Also motion patterns are studied in [5], where floor fields are proposed to determine the probability of moving from one location to another. The main idea is to learn global motion patterns and participants of the crowd are then assumed to behave in a manner similar to the global crowd behavior.

Although these solutions have shown promising results, they impose constraints to the crowd motion. In particular, targets are often assumed to behave in a similar manner, in a such way that all of them have to follow a same motion pattern, consequently, trajectories not following common patterns are penalized. Apparently, this constraint works well in extremely crowded scenes, such as in some religious or sport events, where the movement of individuals within the crowd is restricted by others and by the scene structure as well. Thus, a single object can be tracked by the crowd motion because it is difficult, if not impossible, to move against the main trend. However, the mentioned methods are hard to apply in scenarios where individuals can move in different directions. Besides, some of these methods include other additional constraints, for example, in [100] Rodriguez *et al.* use a limited descriptive representation of target motion by quantizing the optical flow vectors into 10 possible directions. Such a coarse quantization limits tracking to only few directions. Also, the *floor fields* [5] used by Ali *et al.* impose how a pedestrian should move based on scene constraints, which results in only one single direction at each spatial location in the video.

In addition to these solutions based on exploiting global level information about motion patterns to impose constraints to tracking algorithms, similar ideas have been proposed using crowd density measures. In [55], Hou *et al.* use the estimated number of persons in the detection step, which is formulated as a clustering problem with prior knowledge of the number of clusters. This attempt to improve person detection in crowded scenes includes some weaknesses. At least two problems might incur: Firstly, the idea of detection by feature clustering can be only effective in low crowded scenes. It is not applicable in very crowded cases because of the spatial overlaps that make delineating individuals a difficult. Secondly, using the number of people as a crowd measure has the limitation of giving only global information of the entire image and discarding local information about the crowd.

We therefore resort to the crowd density measure introduced in the previous Chapter, in which local information at pixel level substitutes a global number of people per frame, this solution is indeed more appropriate as it enables both the detection and the location of potentially crowded areas. To the best of our knowledge, only one work [101] has investigated this idea. In the referred work, a system which introduces crowd density information into the detection process is proposed. Using an energy formulation, Rodriguez *et al.* [101] show how it is possible to obtain better results than the baseline method [40]. Although it

is a significant improvement of multi-target tracking in crowded scenes, our concern about the referred work is the use of confidence scores from person detection as input to the density estimation. This means the detection scores are used twice, to detect persons and then to estimate crowd density maps which does not introduce any complimentary information in the process. In addition, the proposed crowd density map in [101] involves a training step with large data. Thus, human-annotated ground truth detections are required, and the system is not fully automatic.

In contrast to the previous work, we intend to demonstrate in this Chapter, how it is possible to enhance detection and tracking results using fully automatic crowd density maps that characterize the spatial and temporal variations of the crowd. Compared to the prior works, our approach does not depend on any learning step, and does not impose any direction to the crowd flow. Participants of the crowd are not supposed to behave in a manner similar to the global crowd behavior. It models the crowd in a temporally evolving system, which implies in each space-time location of the video a large number of likely movements.

This additional information is incorporated in a detection and tracking framework: First, the proposed space-time model of crowd density is used as a set of priors for detecting persons in crowded scenes, where we apply the deformable part-based models that has been proposed in [40]. Also, we design a correction filter based on the aspect ratio and the perceived height of a person in order to deal with false positive detections of inappropriate size. Since human detection is a key step in automatic video surveillance, improving the detection results can deeply affect many applications. One application that could be carried out after performing reliable detection is person tracking. To illustrate that, our proposed approach is extended to tracking using a Probability Hypothesis Density (PHD) filter.

The remainder of the Chapter is organized as follows: In the next Section, we introduce the human detector we use. In Section 6.4, we explain how to use the local crowd density measure together with a correction filter in order to improve the detection results. In Section 6.5, an extension of the detection results to tracking is presented. A detailed evaluation of our work follows in Section 6.6. Finally, we briefly conclude.

6.3 Human detection using Deformable Part Based-Models

Human detection is a common problem in computer vision as it is a key step to provide semantic understanding of video data. Accordingly, it has been studied intensively and different approaches have been proposed (e.g. [26], and [40]) which are often gradient-based. In most of the proposed methods, the problem is formulated via binary sliding window classification, where an image pyramid is built and a fixed window size is scanned at all locations and scales to localize individuals.

In this context, the deformable part-based models [40] has recently shown excellent performance. It is an enriched version of Histograms of Oriented Gradients (HoG) [26], that

achieves much more accurate results and marks the current state-of-the-art. The detector uses a feature vector over multiple scales and a number of smaller parts within a Region of Interest (RoI) to get additional cues about an object (see Figure 6.1).

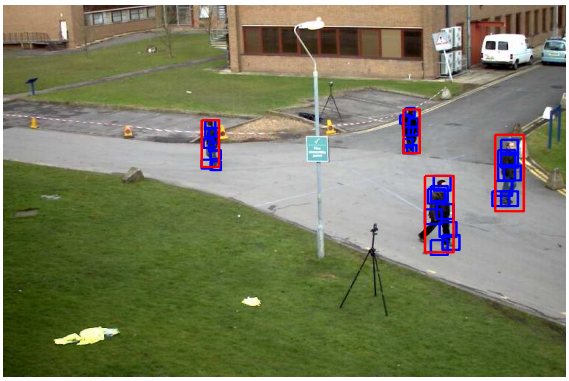


Figure 6.1: Exemplary human detections using the part-based models [40]: Blue boxes describe object parts which also contribute to the overall detection (red).

In this framework, an object hypothesis specifies the location of each filter in a feature pyramid $z = (p_0, ..., p_n)$ with $p_i = (x_i, y_i, l_i)$ as the position and level of the i-th filter. The detection score is given as the score of all filters minus a deformation cost plus a bias b:

$$score(p_0, ..., p_n) = \sum_{i=0}^{n} F_i' \cdot \Psi(H, p_i) \quad - \sum_{i=1}^{n} d_i \cdot \Psi_d(dx_i, dy_i) + b \tag{6.1}$$

with (dx_i, dy_i) as the displacement of the i-th part relative to its anchor position and $\Psi_d(dx, dy)$ as deformation features weighted by the vector d_i.

In this work we use the implementation from [49] which is trained on samples of the INRIA and PASCAL person datasets. The output of the detector is a set of RoIs for a given detection threshold. These must then be processed by an additional non-maximum suppression (NMS) step which in the baseline method essentially maintains regions with high detection scores while removing detections overlapping with these more than a given threshold.

While human detection using the deformable part-based models has become a quite popular technique, its extension to crowded scenes has limited success. In fact, the density of people substantially affects their appearance in video sequences. Especially in dense crowds, people occlude each other and only some parts of each individual's body are visible. Therefore, accurate human detection in such scenarios with frequent occlusions and high interactions among the targets remains a challenge.

To improve the detection performance in crowded scenes, some methods (e.g. [101],

and [8]) rely only on head detections and discard the rest of the body. This is less error-prone but also focuses on a smaller amount of information characterizing a human. Although improved accuracy can be obtained using these solutions, the large amount of partial occlusions in videos of high dense crowds still present big challenges to such detection methods. In order to adapt the detector to these situations, it is important to include additional information about crowds in the scene.

6.4 Integration of geometrical and crowd context constraints into human detector

In this Section, we present our proposed extension of human detection algorithm described in Section 6.3 to crowded scenes. As a major improvement, we propose a variation of the standard non-maximum suppression (NMS) by using the crowd density measure presented in Chapter 5 to improve human detection performance in crowds. In addition, some geometrical constraints are introduced in a first filtering step to remove false positive detections. The remainder of this section is organized as follows: First, the geometrical constraints are defined in a filtering step (Section 6.4.1). Then we present our proposed density-based NMS in crowd-context constraints (Section 6.4.2). In Section 6.4.3, a summary of our proposed integration algorithm is given.

6.4.1 Geometrical Constraints

Due to the part-based nature of the used human detector, it is possible that certain human parts which actually lie on *different* persons are matched together in *one* candidate RoI which then comprises all of them (highlighted in yellow in Figure 6.2 (a)) or that a region is chosen even though it is much too large to contain a human (shown in red in Figure 6.2 (a)). If the score of such detection is higher than the scores of the individual objects' detections, the NMS step will keep it instead of the correct individual detections which might otherwise be recognized. Accordingly, in this case a false positive detection and a number of missed detections are generated which decrease the detection performance. We propose to overcome this problem by applying an outlier filtering step that could ensure that wrong detections will not persist. In the following we define geometry-based pre-filters in order to filter out inaccurate detections of inappropriate size. The design of geometrical correction filters is based on two constraints:

Filtering detections according to the perceived height:

Intuitively, the perceived size of persons in a given image is affected by perspective distortions. The effects of these distortions can be simply explained by the fact that persons far away from the camera appear smaller than closer ones, which makes any detected RoI for persons farther away account for a smaller portion compared to closer persons. Given a set of candidate RoIs $\mathcal{D}_k = \{d_1^k, ..., d_{n_k}^k\}$ at frame k, d_j^k denotes the j^{th} detection at this frame

(a) (b)

Figure 6.2: Exemplary effects of the proposed correction filters on a frame from PETS 2009 dataset [41]: (a) detections without filtering, (b) filtering according to aspect ratio and perceived height. While the unfiltered detections might include too large candidates (red) and also detections comprising several persons at correct height (yellow), the aspect ratio and perceived height allow removing most of them.

and is defined as $d_j^k = \{x_j^k, y_j^k, w_j^k, h_j^k\}$, where (x_j^k, y_j^k) denotes the upper left position of the RoI d_j^k and w_j^k, h_j^k the respective width and height. Following [53] we assume the relationship between a person's position and his perceived height to be:

$$h_j^k = \alpha_{k-1} \cdot y_j^k + \beta_{k-1}, j \in \{1...n_k\} \tag{6.2}$$

where α_{k-1} and β_{k-1} parameters are computed using a standard regression from all accepted detections $\{\mathcal{D}_1, ..., \mathcal{D}_{k-1}\}$ and updated at each frame.

Filtering detections according to the aspect ratio:

While the height of a candidate RoI gives already valuable information about the likelihood of human presence, it does not always identify detections comprising multiple persons at once. Accordingly, we propose to also use the aspect ratio as a correction measure. γ_{k-1} computed over all accepted detections $\{\mathcal{D}_1, ..., \mathcal{D}_{k-1}\}$ is defined as:

$$\gamma_{k-1} = median \left\{ \frac{w_j^i}{h_j^i} \right\}_{1 \leq i \leq (k-1), 1 \leq j \leq n_i} \tag{6.3}$$

These proposed correction filters use the previous detections of the video to predict the height and the ratio of a new detection candidate, allowing the algorithm to operate on-line without any previous learning step. By applying these two geometrical filters simultaneously, a detection candidate is accepted only if it fits the aspect ratio and the height according to the y-coordinate of its center.

As the used NMS step is greedy and overlap-oriented, it is now possible to filter out

an unlikely large or small region and to detect other objects in the same area which would have been suppressed otherwise. An example of this correction filters can be seen in Figure 6.2 (b) where false positive detections from the previous images are suppressed.

6.4.2 Crowd Context Constraint:

The usage of detection thresholds in many human detectors can cause difficulties in real-world applications. Beforehand it is not always clear to the user how to adapt the algorithm to a new scene and how to choose the threshold value. While lower values usually increase the number of detections and allow recognizing more persons, they also increase the number of false positives. On the other hand, higher thresholds only detect more reliable candidate regions but might cause the detector to miss some people in the scene. This is especially difficult in heterogeneous scenes with crowded and non-crowded regions and is due to the fact that high crowd scenes present many challenges that are not present in low-crowd scenes. These include the large number of persons, small target size, occlusions because of inter-object interactions. The impact of these difficulties on the detection results is highly dependent on the crowd size i.e. the higher crowd density, the more difficult to detect persons. As a result, low detection thresholds would be suitable in crowded scenes and higher values ensure less false positives in non-crowded spaces. It is therefore desirable to find a way of automatically setting the detection threshold τ according to the probability that people are present in a certain position of the image. As explained in Chapter 5, crowd density maps could exactly provide this information. Therefore, we propose to use them in order to adjust the detection threshold according to the local density.

In the detection step, we obtain a set of candidate RoIs in a video sequence of N frames $\{I_1, ..., I_N\}$ for a given threshold τ: $\mathscr{D}(\tau) = \{\mathcal{D}_1, ..., \mathcal{D}_N\}$, where $\mathcal{D}_k = \{d_1^k, ..., d_{n_k}^k\}$ denotes the set of detections at frame k. Using a pre-defined range of detection thresholds given by an upper/lower boundary τ_{max}/τ_{min}, we apply the following method of computing a suitable value automatically:

$$\tau_{dyn} = \tau_{min} + (\tau_{max} - \tau_{min}) \cdot \hat{C}_k(d_j^k), j \in \{1...n_k\} \tag{6.4}$$

with

$$\hat{C}_k(d_j^k) = \frac{\sum_{p=0}^{h_j^k-1} \sum_{q=0}^{w_j^k-1} C_k(x_j^k + p, y_j^k + q)}{w_j^k \cdot h_j^k} \tag{6.5}$$

as the average crowd density value of detection d_j^k.
To obtain the dynamic threshold τ_{dyn} for every candidate d_j^k in $\mathscr{D}_{min}$, the average crowd density $\hat{C}_k(d_j^k)$ is computed as in (6.5) and inserted into (6.4) for all regions.

6.4.3 Summary of the integration algorithm

Algorithm 3 shows in pseudo-code an overview of our proposed human detection algorithm in crowds by integrating geometrical and crowd context constraints into the state-of-the-art human detector.

Algorithm 3: Proposed Human Detection in Crowds

Input:

- $\mathscr{I} = \{I_k\}_{1 \leq k \leq N}$, N frames of a given video sequence V and their corresponding crowd density maps $\mathscr{C} = \{C_k\}_{1 \leq k \leq N}$.
- $\mathscr{D} = \{\mathcal{D}_k\}_{1 \leq k \leq N}$: a set of preliminary candidate detections and their corresponding scores $\mathscr{S} = \{\mathcal{S}_k\}_{1 \leq k \leq N}$.

Output: Selected detections $\mathscr{D}''$
Initialize: Set $(\alpha_0, \beta_0, \gamma_0)$ parameters to $-\infty$
for $k = 1$ **to** N **do**
$\mathcal{D}_k = \{d_1^k, ..., d_{n_k}^k\}$, $\mathcal{S}_k^k = \{s_1^k, ..., s_{n_k}^k\}$

1. **Filtering**:
   if $(\alpha_{k-1} = -\infty)$
   $\mathcal{D}'_k \leftarrow \mathcal{D}_k$, $\mathcal{S}'_k \leftarrow \mathcal{S}_k$
   else
   $(\mathcal{D}'_k, \mathcal{S}'_k) \leftarrow$ Apply filtering $(\mathcal{D}_k, \mathcal{S}_k, \alpha_{k-1}, \beta_{k-1}, \gamma_{k-1})$
   end if
2. **nms-based-density:**
   $\mathcal{D}'_k = \{d'^k_1, ..., d'^k_{m_k}\}$, $\mathcal{S}'_k = \{s'^k_1, ..., s'^k_{m_k}\}$
   - $Index_1^k \leftarrow$ Sort confidence scores $\mathcal{S}'_k$
   - **for** each position $i \in Index_1^k$ **do**
     Compute ratio of overlap ϑ_{ij}^k between detections at
     $Index_1^k(i)$ and at $Index_1^k(j)$, $(i+1) \leq j \leq m_k$
     end for
   - $Index_2^k \leftarrow$ Remaining index after removing all overlapped detections more than a certain threshold $\Delta_o = 0.5$
   - $C_k \leftarrow$ Normalize Density Map C_k to $[0...1]$
   - For all pixels $x \in I_k$, compute detection thresholds using a predefined range of detection thresholds $[\tau_{min}...\tau_{max}]$ and the normalized C_k
   - $Index_F^k = \{\}$
   - **for** $c = 1$ **to** length $(Index_2^k)$ **do**
     $\tau_{dyn}(d'^k_{Index_2^k(c)}) \leftarrow$ average of detection threshold values of all pixels belonging to the RoI
     if $(s'^k_{Index_2^k(c)} \geq \tau_{dyn})$, then $Index_F^k \leftarrow \{Index_F^k, c\}$
     end for
   - $\mathcal{D}''_k \leftarrow \mathcal{D}'_k\{Index_F^k\}$
3. $(\alpha_k, \beta_k, \gamma_k) \leftarrow$ Update Filtering Parameters $(\{\mathcal{D}''_l\}_{1 \leq l \leq k})$

end for

The implementation of this algorithm can be effectively done as follows: Firstly, a set of

candidate RoIs $\mathscr{D}$ is computed for the minimal detection threshold τ_{min}. This set contains all possible detections which can be extracted for the given threshold range $[\tau_{min}...\tau_{max}]$. To filter out inaccurate detections of inappropriate size, the two proposed geometrical filters are applied. A detection d_j^k= (x_j^k, y_j^k, h_j^k, w_j^k) is accepted only if it fits the predicted ratio and height with error less than certain thresholds (Δ_γ, Δ_h) i.e. only if ($(w_j^k/h_j^k) \leq \gamma_{k-1} \pm \Delta_\gamma$) and ($h_j^k \leq \widetilde{h}_j^k \pm \Delta_h$), where $\widetilde{h}_j^k$ denotes the predicted height of the bounding box, computed from (6.2).

After applying these two geometrical filters, we obtain a set of new detections $\mathcal{D}_k'$ and their corresponding scores $\mathcal{S}_k'$. At this stage, we often get multiple overlapping detections, thus we use a greedy procedure for eliminating repeated detections. It proceeds by sorting the detections $\mathcal{D}_k'$ according to their corresponding scores and greedily selecting the highest scoring ones while skipping overlapped detections that are covered by more than 50% by a bounding box of a previously selected detection. The following step consists of thresholding the remaining detections using the computed dynamic threshold according to the crowd density. Finally, the filtering parameters α_k, β_k, and γ_k are updated according to the new selected detections $\mathcal{D}_k''$. In the following, $\mathcal{D}_k$ denotes the selected detections.

6.5 Tracking-by-detection using Probability Hypothesis Density

A huge number of different approaches have been proposed for human tracking which are usually based on using multiple single-object filters. While traditional methods for multi-object tracking as e.g. Probabilistic Data Association [115] or Multiple Hypothesis Tracking [10] can be used, these rely on the estimation of a number of single-object states and thus suffer from an exponential computational effort in the number of persons in the image.

As a remedy, Mahler proposed the Probability Hypothesis Density (PHD) filter [82] which is based on Finite Set Statistics (FISST) and models a joint multi-object state for all objects in the current time step using a set representation. Although a multi-object Bayes tracker is in theory computationally intractable, the PHD filter relies on an approximation which propagates only the first moment of the desired multi-object posterior and its complexity is reduced to $\mathcal{O}(mn)$ with m being the number of observations and n as the number of targets in the scene [81].

For implementation, we use a Gaussian-Mixture Probability Hypothesis Density (GM-PHD) filter [126] which similarly to the well-known Kalman Filter [63] assumes a linear motion model and expresses the PHD function $\Theta(\mathbf{x})$ at time step k as a mixture of Gaussians with their respective mean and covariance values $\mu_k^{(i)}, \Sigma_k^{(i)}$:

$$\Theta_k(\mathbf{x}) = \sum_{i=1}^{J_k} w_k^{(i)} N(\mathbf{x}; \mu_k^{(i)}, \Sigma_k^{(i)}) \tag{6.6}$$

This filter models the PHD function $\Theta(\mathbf{x})$ at time step k as a mixture of Gaussians and propagates them in an estimation step from the previous state **x'** according to the object motion model $f(\mathbf{x}|\mathbf{x'})$. A survival probability $p_S(\mathbf{x'})$ can account for exit points in a scene. Additionally, birth distributions $N_b(\mathbf{x})$ are added in the estimation step for all detections in order to account for new objects:

$$\Theta_{k|k-1}(\mathbf{x}) = N_b(\mathbf{x}) + \sum_{i=1}^{J_k} p_S(\mathbf{x'}) \cdot f(\mathbf{x}|\mathbf{x'}) \cdot \Theta_{k-1|k-1}(\mathbf{x'}). \tag{6.7}$$

In the following correction step, the PHD function is then adapted according to the currently received measurement set $\mathcal{D}_k$:

$$\begin{aligned} \Theta_{k|k}(\mathbf{x}) \quad &= (1 - p_{det}(\mathbf{x})) \cdot \Theta_{k|k-1}(\mathbf{x}) + \\ &\int \frac{p_{det}(\mathbf{x}) \cdot L_{d_i^k}(\mathbf{x}) \cdot \Theta_{k|k-1}(\mathbf{x})}{\mathcal{C} + \int p_{det}(\mathbf{x}) \cdot L_{d_i^k}(\mathbf{x}) \cdot \Theta_{k|k-1}(\mathbf{x}) d\mathbf{x}} \mathbf{d} d_i^k \end{aligned} \tag{6.8}$$

where the detection probability p_{det} and the clutter rate C characterize the used human detector, and $L_{d_i^k}(\mathbf{x})$ is the likelihood for a given measurement d_i^k and a state $\mathbf{x}$.

In the used GM-PHD filter, this correction step is performed by generating $(J_{k-1} + |\mathcal{D}_k|) \cdot (1 + |\mathcal{D}_k|)$ new Gaussian distributions. While their mean and covariance values are chosen according to the position of the respective state and detection, the weights of the corrected curves are computed as follows:

$$w_k^{[j]}(d_i^k) = \begin{cases} (1 - p_{det}) \cdot w_{k|k-1}^{[j]}, & \text{no detection} \\ \frac{p_{det}(\mathbf{x}) \cdot L_{d_i^k}(\mathbf{x}) \cdot w_{k|k-1}^{[j]}}{\mathcal{C} + \int p_{det}(\mathbf{x}) \cdot L_{d_i^k}(\mathbf{x}) \cdot w_{k|k-1}^{[j]} d\mathbf{x}}, & \text{for } d_i^k \in \mathcal{D}_k \end{cases} \tag{6.9}$$

In order to keep the overall number of Gaussians at a suitable level, merging and pruning procedures as proposed in [20] are carried out. After this step, object extraction is done by reporting hypotheses with a weight of $\Theta(\mathbf{x}) > T_{extract}$ (usually set to $T_{extract} = 0.5$).

For object identification, we use a feature-based label tree extension as proposed in [34]. This extension uses image information to distinguish objects and is especially useful in cases of near objects and occlusions which are given in our scenarios. From (6.9), it can be seen that the PHD filter is sensitive to missed detections. In case, no current detection confirms a state estimate, its weight is reduced by the constant factor $(1 - p_\Theta)$. Should it fall below $T_{extract}$, it will not be reported and the corresponding track will not be continued in this frame. It is therefore important to ensure a high detection probability of the human detector used as input to the PHD filter.

6.6 Experimental Results

6.6.1 Datasets and Experiments

The proposed approach is evaluated within challenging crowd scenes from multiple video datasets. In particular, we select some videos from PETS 2009 [41], UCF dataset [4], and the data-driven crowd analysis dataset [103]. These videos are annotated for all frames using Viper [85] (except for UCF-879 where the annotation comprises only the first 200 frames).

To demonstrate the effectiveness of the proposed detection algorithm, we compare our results to the baseline algorithm [40]. In particular, two detection thresholds (as τ_{min} and τ_{max}) are tested for the baseline algorithm, whereas the proposed method uses a dynamically chosen threshold between these values according to the crowd density. Additional tests are conducted to assess the impact of the correction filters.

For quantitative evaluations, we use the CLEAR metrics proposed in [121]. These are split in two parts: the Multi-Object Detection Accuracy (MODA, N-MODA) and the Multi-Object Detection Precision (MODP, N-MODP). The first step in computing the metrics for a set of detection RoIs $\mathcal{D}_k = \{d_1^k, ..., d_{n_k}^k\}$ and the corresponding ground truth detections $\phi_k = \{\varphi_1, ..., \varphi_{l_k}^k\}$ is to match both sets in order to identify which ground truth detections have been found by the detector. Taking a spatial overlap ratio between all pairs as input, we use the well-known Hungarian algorithm for this assignment. As proposed in [121], a threshold of 0.2 for the overlap ratio prevents assignments between badly matching pairs. Once the assignment for all frames is done, MODP (t) is computed as the summed and normalized overlap ratio between all assigned pairs in the image:

$$MODP(t) = \frac{OverlapRatio}{N_{mapped}^t} \tag{6.10}$$

with N_{mapped}^t as the number of assigned object regions in frame t. N-MODP then gives normalized localization results for the entire sequence using the MODP values of all frames:

$$N - MODP(t) = \frac{\sum MODP(t)}{N_{frames}} \tag{6.11}$$

The N-MODA metric measures the accuracy aspect of the system's performance over the video sequence and is essentially a normalized sum of false positives and missed detections:

$$N - MODA(t) = 1 - \frac{\sum_{i=1}^{N_{frames}} (m_i + f_{p_i})}{\sum_{i=1}^{N_{frames}} N_G^i} \tag{6.12}$$

with m_t as the number of missed objects, f_{p_t} as the number of false positives (clutter) and N_G^t as the number of ground truth objects in frame t. Both N-MODP and N-MODA illustrate best performance results by a value equal to 1 while lower values indicate worse

performance.

Finally, to evaluate the tracking performance, we use the OSPA-T distance proposed in [99]. This metric is mathematically rigorous and defined on the space of finite sets of tracks. Extending OSPA metric [107] by Schuhmacher *et al.* it integrates the distance between ground truth tracks and estimated tracks (position error) as well as errors in the number of objects (cardinality error) and labeling errors into the assessment of a tracker's performance.

Let $X_k = \{x_1, ..., x_m\}$ and $Y_k = \{y_1, ..., y_n\}$ be the existing ground truth position sets and the multi-object state estimates (also in set formulation) produced by the tracking system at timestep k. The OSPA distance between X and Y is then defined as:

$$D_{p,c}(X,Y) = \left[\frac{1}{n}\left(\min_{\pi\in\Pi_n}\sum_{i=1}^{m}(d_c(x_i, y_{\pi(i)}))^p + (n-m)\cdot c^p\right)\right]^{\frac{1}{p}} \tag{6.13}$$

with

- $d_c(x,y) = \min(c, d(x,y)$ as the so-called *cut-off distance* between two tracks with $c > 0$
- $d(x,y)$ as the *base distance* between two tracks which also takes into account labeling errors
- m, n as the cardinalities of the two track sets
- π as the permutation (from the set of possible point assignments Π_n) of length $m \leq n$ with elements $\{1, 2, ..., n\}$ minimizing the error
- $1 \leq p < \infty$ as the OSPA metric order

To evaluate a multi-object state estimate using the OSPA-T metric, it is first necessary to assign all estimated tracks to their best-fitting ground truth counterpart. This is done using the Hungarian algorithm in order to obtain an optimal point assignment between the two sets. For every timestep, the OSPA distance (6.13) is then computed. The first term represents the spatial distance between the assignments (using a maximally possible penalty value c) while the second term accounts for cardinality errors in the estimate. The sum of both terms is normalized using the p-th order average.

6.6.2 Results and Analysis

For the detection part, the results using static detection thresholds τ_{min}, τ_{max} (baseline method) are compared to the proposed dynamic threshold $\tau_{dyn} \in \{\tau_{min}...\tau_{max}\}$ in Table 6.1. We set τ_{min} to (-0.5) and τ_{max} to (-1.2), these values have been found empirically suitable for lowly resp. highly crowded scenes. The first column of this Table shows that

using (-0.5) as detection threshold does not provide satisfactory results, and by decreasing the threshold to (-1.2) in the second column, the results are even worse. That is why, we consider that using adaptive threshold based on crowd context is more appropriate method. As shown in the third column, the automatic choice of the detection threshold already gives better results than both configurations of the baseline method. Regarding the final results (in the last column), the proposed system using a dynamically chosen detection threshold and correction filtering gives the best results for all test videos. These results demonstrate that integrating both proposed steps (filtering and dynamic threshold) into human detector performs favorably better than implementing them separately which justifies that filtering has to be performed first to suppress false detections and to emphasize correct ones. Again, the choice of the feature detector in general does not seem critical to the performance, expect slight improvement using FAST compared to other features. This due to the fact, that for tested videos both Crowd and No Crowd regions are considered.

Although the PETS 2009 sequences provide all the same view (View 1), they still pose different problems to the detector. Changing lighting conditions, shadows and different crowd densities between the test sequences are challenging and in all cases, the proposed method improves the detection results over the baseline method. Due to the higher crowd density and the tilted camera view, the UCF-879 sequence is even more challenging. However, the proposed method still enhances the detection considerably compared to the baseline method. For the INRIA 879-38_I sequence, the camera view is almost completely downward and people are walking very near to the camera which changes their aspect ratio considerably for different positions. Additionally, for this specific perspective, many detection candidates comprising the head of one person and the body of another are generated. As the correction filter does not apply a prior-knowledge about the shape of a person but is only trained on previous detections, it is misled in this situation. Accordingly, in this special case its contribution is smaller.

Figure 6.3 shows exemplary visual results which also indicate the performance increase by the proposed method. Since the part-based model represents the current state-of-the-art detector, we consider extending it to operate in crowded scenes and improving its performance is a substantial contribution. As an advantage of our method the proposed extensions do not need a previous learning phase and can be applied on-line.

For tracking, the results of test sequences which are generated using the same tracker configuration for all tests on every video to ensure comparability are shown in Table 6.2. Generally, the results of the proposed method using a dynamical detection threshold and correction filtering are better compared to the baseline method. The gain is especially high for the sequences PETS S1.L2.14-31 and INRIA-879-42_I but an overall major improvement can be seen in all videos.

These results are consistent with our expectations as the tracker relies on improved detections. And lower clutter and more accurate detections both improve the tracking. OSPA-T values change more between different feature types than the MODA/MODP values due

to the filtering effect of the PHD tracker. As the tracker can deal with clutter and also missed detections to a certain degree, detection improvements enhance the tracking performance but not all of them have the same effect. So it is possible that the tracking results may vary over different feature types, although they may generate similar MODA/MODP results.

The OSPA-T metric for different configurations over a complete scene (PETS S1.L2.14-31) is shown in Fig. 6.4 (a). For this scene with hard lighting conditions and medium crowd density, the detection performance is increased considerably by the proposed method. The diagram shows that the tracking performance of our method is mostly better than using the baseline algorithm. Visual examples are given in Fig. 6.4 (b)-(e) where it can be seen that our method is visibly able to track objects for a longer time than the baseline method and also maintains more tracks than the standard method.

6.7 Conclusion

In this Chapter, we proposed an extension of the part-based human detection to crowded scenes by incorporating local crowd density and geometrical correction filters in the non-maximum suppression step and used the detection results for human tracking. By means of automatically estimated crowd density maps, the detection threshold of a human detector is adjusted according to the scene crowd context. In order to cope with false positive detections of inappropriate size, dynamically-learning correction filters exploiting the aspect ratio and the perceived height of detections are proposed. None of the proposed extensions need a training phase and both can be applied on-line. An extensive evaluation on several datasets demonstrates the advantages of incorporating local crowd density into the detection and tracking process.

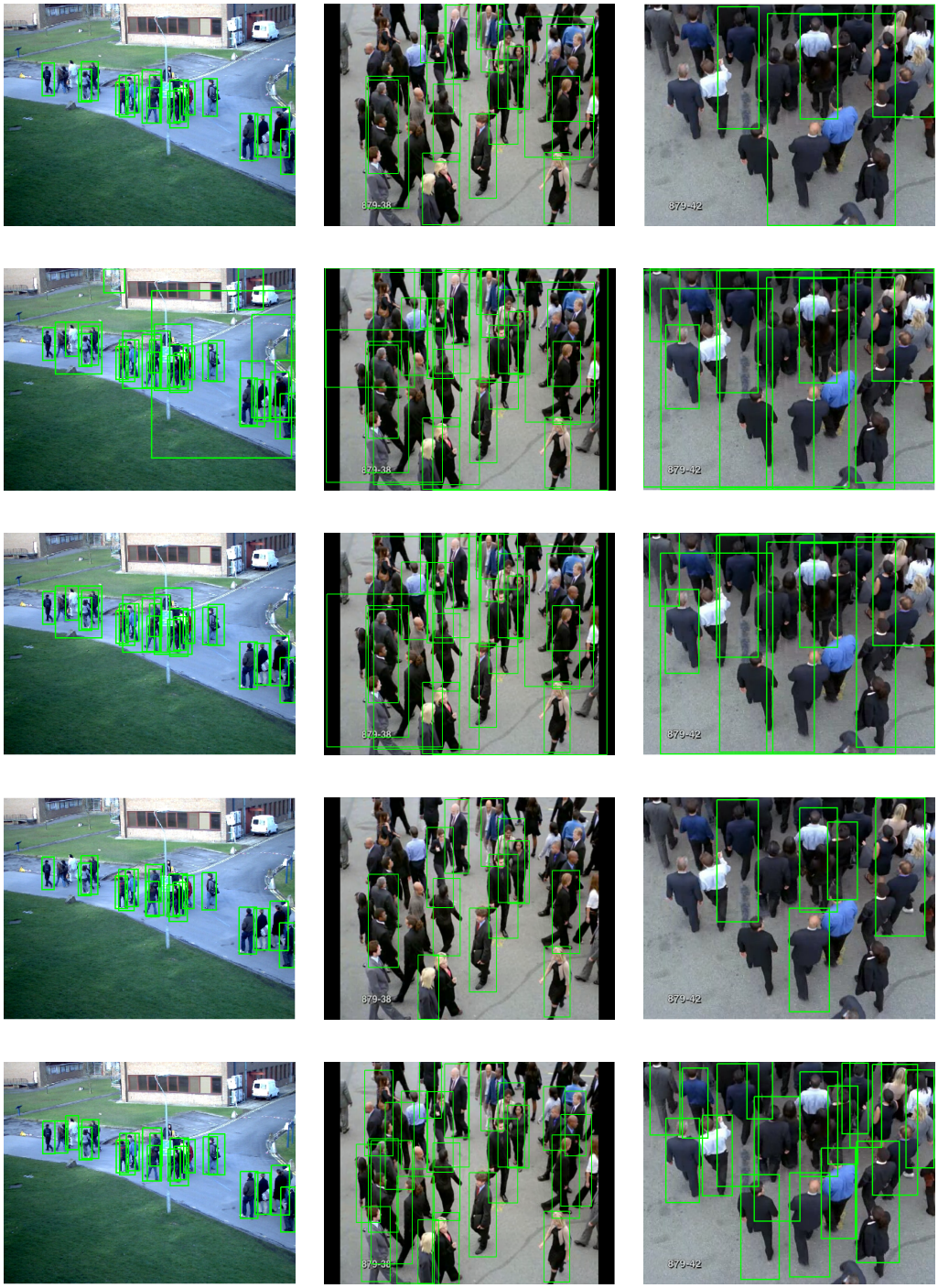


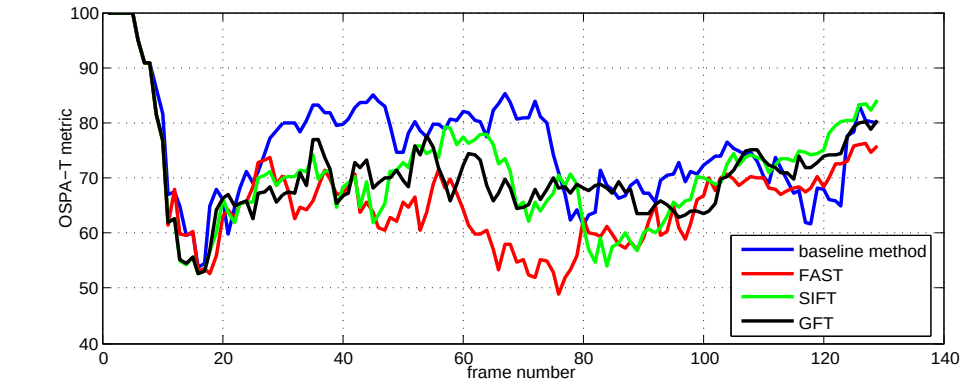
Figure 6.3: Exemplary visual results show how a crowd-sensitive threshold increases the detection performance compared to the baseline method while the proposed algorithm using an additional correction filter enhances the results further: (a) baseline algorithm at τ_{min}, (b) baseline algorithm at τ_{max}, (c) dynamically chosen τ, (d) filtered detections (e) proposed method using dynamically chosen τ and correction filter according to aspect ratio and perceived height. From Top to bottom: Frames from PETS 2009, UCF 879, and INRIA 879-38_I. For PETS and UCF, the proposed method generates more accurate detections and less clutter compared to the baseline method. Results for INRIA are also visibly better, but due to the camera view the effect of the correction filter is small.

sequence name	$\tau_{min} = -0.5$	$\tau_{max} = -1.2$	$\tau_{dyn} \in \{\tau_{min} ... \tau_{max}\}$	Filtering	τ_{dyn} + Filtering
PETS S1.L1.13-57 (FAST):			0.59 / 0.59		**0.63 / 0.63**
PETS S1.L1.13-57 (SIFT):	0.48 / 0.65$^{(*)}$	0.36 / 0.57 $^{(*)}$	0.59 / 0.60	0.48 / 0.66	0.61 / 0.63
PETS S1.L1.13-57 (GFT):			**0.60 / 0.60**		0.62 / 0.63
PETS S1.L1.13-59 (FAST):			**0.60 / 0.67**		0.60 / 0.68
PETS S1.L1.13-59 (SIFT):	0.56 / 0.68$^{(*)}$	0.25 / 0.61$^{(*)}$	**0.60 / 0.67**	0.56 / 0.69	0.60 / 0.68
PETS S1.L1.13-59 (GFT):			0.59 / 0.67		**0.61 / 0.68**
PETS S1.L2.14-31 (FAST):			**0.40 / 0.59**		**0.47 / 0.63**
PETS S1.L2.14-31 (SIFT):	0.33 / 0.63$^{(*)}$	0.09 / 0.57$^{(*)}$	**0.40 / 0.59**	0.32 / 0.65	**0.47 / 0.63**
PETS S1.L2.14-31 (GFT):			**0.40 / 0.59**		**0.47 / 0.63**
PETS S2.L3.14-41 (FAST):			**0.34 / 0.56**		**0.35 / 0.57**
PETS S2.L3.14-41 (SIFT):	0.29 / 0.54$^{(*)}$	0.04 / 0.56 $^{(*)}$	0.34 / 0.54	0.29 / 0.54	0.35 / 0.55
PETS S2.L3.14-41 (GFT):			0.34 / 0.54		0.36 / 0.55
UCF-879 (FAST):			0.41 / 0.55		**0.59 / 0.58**
UCF-879 (SIFT):	0.44 / 0.58$^{(*)}$	0.34 / 0.54$^{(*)}$	0.42 / 0.55	0.41 / 0.62	0.57 / 0.58
UCF-879 (GFT):			**0.43 / 0.55**		0.58 / 0.58
INRIA-879-42_I (FAST):			**0.35 / 0.55**		**0.42 / 0.47**
INRIA-879-42_I (SIFT):	0.27 / 0.54$^{(*)}$	0.06 / 0.55 $^{(*)}$	**0.35 / 0.55**	0.20 / 0.42	0.38 / 0.45
INRIA-879-42_I (GFT):			**0.35 / 0.55**		0.41 / 0.44

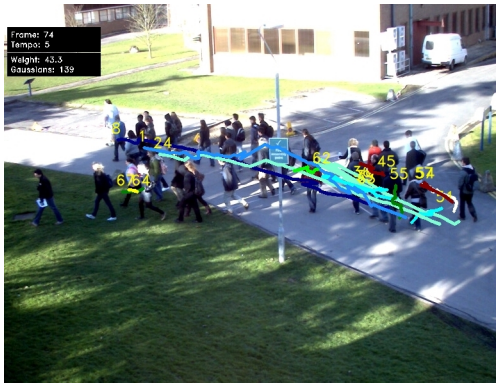
Table 6.1: N-MODA / N-MODP results for three different feature types used in the crowd density estimation (FAST / SIFT / GFT) and for different test videos. Baseline method [40] using a fixed τ marked by $^{(*)}$. Higher values indicate better performance. The proposed system using dynamical detection thresholds and correction filtering is in all cases among the best results while the performance does not change significantly for different feature types.

sequence name	original ($\tau = 0.5$)	proposed method
PETS S1.L1.13-57 (FAST):		63.64
PETS S1.L1.13-57 (SIFT):	65.26$^{(*)}$	62.69
PETS S1.L1.13-57 (GFT):		**61.06**
PETS S1.L1.13-59 (FAST):		**62.36**
PETS S1.L1.13-59 (SIFT):	64.81$^{(*)}$	64.61
PETS S1.L1.13-59 (GFT):		64.05
PETS S1.L2.14-31 (FAST):		**66.39**
PETS S1.L2.14-31 (SIFT):	75.27$^{(*)}$	70.82
PETS S1.L2.14-31 (GFT):		71.00
PETS S2.L3.14-41 (FAST):		87.65
PETS S2.L3.14-41 (SIFT):	88.19$^{(*)}$	88.44
PETS S2.L3.14-41 (GFT):		**87.36**
UCF-879 (FAST):		86.89
UCF-879 (SIFT):	89.92	86.95
UCF-879 (GFT):		**86.46**
INRIA-879-42_I (FAST):		**73.22**
INRIA-879-42_I (SIFT):	81.15$^{(*)}$	75.55
INRIA-879-42_I (GFT):		73.56

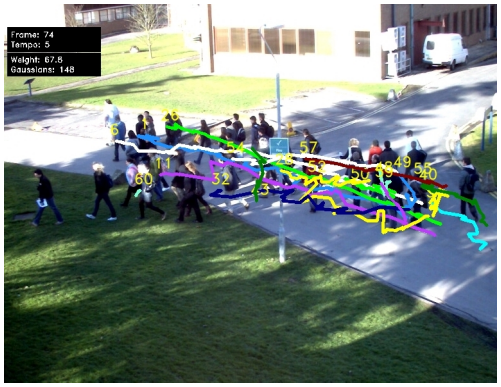
Table 6.2: Averaged OSPA-T values for test sequences and different feature types (FAST / SIFT / GFT). We use a cut-off parameter $c = 100$, $\alpha = 30$ and a distance order of $d = 2$. Lower values indicate better performance. The proposed system using dynamical detection thresholds and correction filtering gives mostly better results than the baseline method. However, due to the filtering effect of the tracking algorithm, the overall improvement changes over different feature types. The improvements are mostly consistent with the detection results (see Table 6.1).



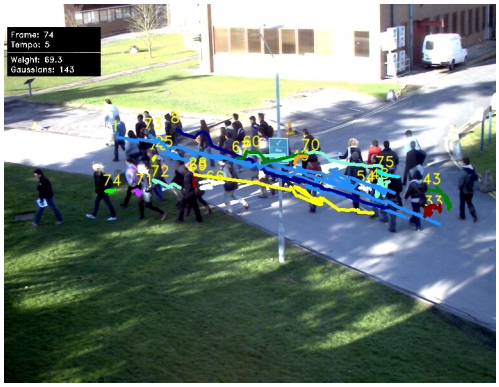
(a)



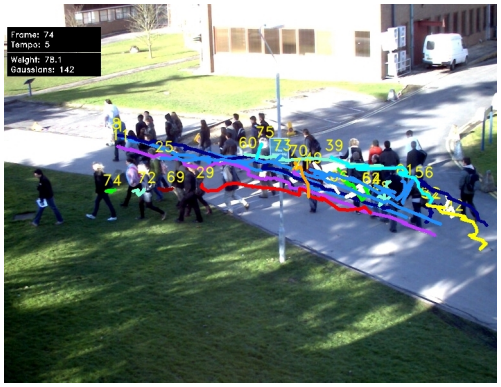
(b)



(c)



(d)



(e)

Figure 6.4: (a) OSPA-T distance over full sequence PETS S1.L2.14-31. Lower values are better. Independent from the feature type, the proposed method shows generally a better performance compared to the baseline method. (b)-(e) Exemplary visual tracking results for this scene. Tracks are visibly maintained longer and more tracks could be established by the proposed method compared to the baseline method. (b) baseline method, (c) proposed method using FAST features, (d) proposed method using SIFT features, (e) proposed method using GFT features

CHAPTER 7

Contextualized Privacy Preservation Filters Using Crowd Density Maps

7.1 Introduction

The widespread growth in the adoption of digital video surveillance systems emphasizes the need for privacy-preservation video analytic techniques. While these privacy aspects have shown big interest in recent years, little importance has been given to the concept of context-aware privacy protection filters. In this Chapter, we specifically focus on the dependency between privacy preservation and crowd density. We show that additional information about the crowd density in the scene can be used in order to adjust the level of privacy protection according to the local needs. According to this additional information cue that consists of modeling time-varying dynamics of the crowd density, the protection level of personal privacy in videos is adapted. Afterwards, a framework for objective evaluation of the contextualized protection filters is proposed.

7.2 Related Works

In recent years, a widespread growth in the adoption of digital video surveillance systems for monitoring buildings and public spaces has been observed. In this context, several concerns have been raised related to the possibility of infringing the privacy rights of the subjects being monitored [90]. At the same time, the adoption of automated methods for the analysis of video surveillance data has raised additional concerns, since algorithms such as face recognition or people re-identification could potentially expose the identity of any individual under video surveillance at any time [109].

Privacy aspects in video surveillance systems have been discussed in different approaches. In [109], [112] and [129] extensive overviews of general requirements such as the need for integrity, confidentiality or access authorization are given. In [14], Cavallaro points out how the ongoing changes towards digital CCTV footage lead to easier storing, transmission and analysis of video data compared to earlier years. This also enables CCTV network operators to choose which analysis tasks have to be run in real-time and which can be done on stored video data as not all tasks have to be carried out in all scenes and contexts. Consequently, [14] proposes to use a privacy-by-design approach in which smart cameras split

the recorded data into a behavioral part and a part containing personal data. From this splitting point on, a video operator can only access the behavioral part while personal data is maintained confidential and only stored in a video archive in order to allow a later access for police and law enforcement agencies (if this is needed and permitted by jurisdiction).

In general, current video surveillance systems either do not implement any mechanism for privacy protection, or they use naïve approaches, for instance uniformly applying simple filters (e.g. masking, Gaussian blur, and pixelization) to some regions of the image which contain privacy sensitive information, such as faces or license plates. The lack of specific methods to detect privacy sensitive regions of interest and to evaluate the amount of privacy protection required in a specific scenario often causes failure in either minimizing the intrusion of the surveillance system or goes against the purpose of the surveillance itself. One big challenge in defining privacy protection policies for video surveillance applications is the identification of the correct trade-off between intelligibility of the video, which should be adequate to the monitoring tasks, and privacy protection itself. Consequently, a number of recent studies have been conducted to propose more adequate systems for privacy protection.

In this perspective, a fundamentally new approach based on the concept of scene-dependent privacy levels has emerged. It is a natural and intuitive idea that a specific human action in a video has to be considered according to the scene context. As a simple example, detection of fireworks in a train station on a normal day would be an unusual and potentially dangerous event but can be mostly considered normal in an outdoor scene on New Year's Eve. A context-dependent approach to privacy protection is described in [6], where image processing and scene understanding techniques are employed to automatically evaluate the context in which video surveillance takes place, in order to apply context-specific privacy rules. This approach is based on scene and object detection algorithms such as bag-of-visual-words, people tracking and gait analysis in order to recognize specific sub-contexts which require the application of different privacy protection rules. In [87], the authors propose another context-aware surveillance system, where the situation within an environment is interpreted by combining a number of contextual information, which are then used to determine an appropriate level of privacy. Six levels of privacy protection ranging from high to low are proposed, and their application is based on the analysis of visual features such as global motion in the scene and detection-based crowd size estimation.

As employed in [87], the crowd size (or more precisely the number of people in the scene) can be an important indication of which events are expected and therefore which privacy level is suitable in the scene. If we take crowd management as an exemplary standard task within the field of video surveillance, video operators need clear visual information in crowded regions. Mainly in case of abnormal events such as potential overcrowding or dangerous motion patterns, a video operator should be able to perceive the maximal information for early detection of unusual situations in large scale crowd to ensure assis-

tance and emergency contingency plan and to decide if an intervention by security forces is needed. At the same time, when there are more people in the scene, a single individual is less perceivable and identifiable. Therefore, it is important in many applications to reduce the privacy filtering level in crowded areas compared to low-crowded areas.

In this Chapter, we propose a system which is able to choose a suitable level of privacy according to a crowd density measure. In particular, we employ the crowd density map (proposed in Chapter 5 because local information at pixel level is more relevant than a global number of persons or a crowd level). Our following objective is to use the estimated density maps in order to build adaptive privacy protection filters, in which the privacy level gradually decreases with the crowd density. As an additional contribution of this Chapter, we identify a framework for objective evaluation, which enables assessing the intelligibility vs. privacy balance based on the performances of state-of-art video surveillance analysis algorithms. In our experiments, we intend to demonstrate that the proposed contextualized privacy protection filters are resistant to local features-based person matching algorithms, which potentially threaten one's individual privacy, while still preserving those visual features which are fundamental for automated crowd analysis tasks such as people detection and counting.

The remainder of the Chapter is organized as follows: Section 7.3 shows how the crowd density information is incorporated into a privacy protection framework which alters the data protection level accordingly. The objective evaluation framework and results using the proposed contextualized filters on different video sequences are given in Section 7.4. Finally, we briefly conclude in Section 7.5.

7.3 Incorporation of Crowd Density Measure in a Privacy Preservation Framework

In this Section, we propose to apply crowd density information for context-aware privacy purposes. In particular, the proposed crowd density measure described in Chapter 5 is employed to adjust the level of privacy protection according to the local needs. The reason behind that is to hide personal information to the video operator without preventing him to be able to identify potential dangerous areas and events. A simple way for that could be to just use crowd density directly as an input to a privacy filter in such way that the obfuscation level directly depends on the density of a given region. This method could substantially decrease the visibility of potentially important information since all crowded areas would be obscured.

Because of that, we restrict the application of privacy preservation filers to some regions of interest, i.e. only regions that contain personal information are obfuscated. These could include face, clothing, skin/hair color or even gait depending on the scene context.

Given this variety and considering that these information is not perceivable under all circumstances (e.g. heavy crowding, different lighting conditions, motion blur, low contrast, low resolution...), in our work we consider head obfuscation as the most visible part of a human in a crowd. However, once a person has left the crowd and is perceived as an isolated subject, more information has to be hidden. This is why in these cases we extend the obfuscated region to the whole body in order to hide details such as clothing or skin color from the viewer.

As a measure for privacy protection, the level of obfuscation is adapted according to the crowd density for the following reasons: Crowds are usually interesting to video operators as they are a common place for crimes or for dangerous overcrowding events. At the same time, people in a crowd exhibit a smaller amount of information to a video operator, thus they do not have to be filtered to the same degree as for isolated people who are entirely visible. We therefore propose to lower the level of privacy protection within a crowded area. The flowchart of the proposed contextualized privacy protection filers is shown in Figure 7.1. In the following, we describe our system components: RoIs detection and adaptive filters.

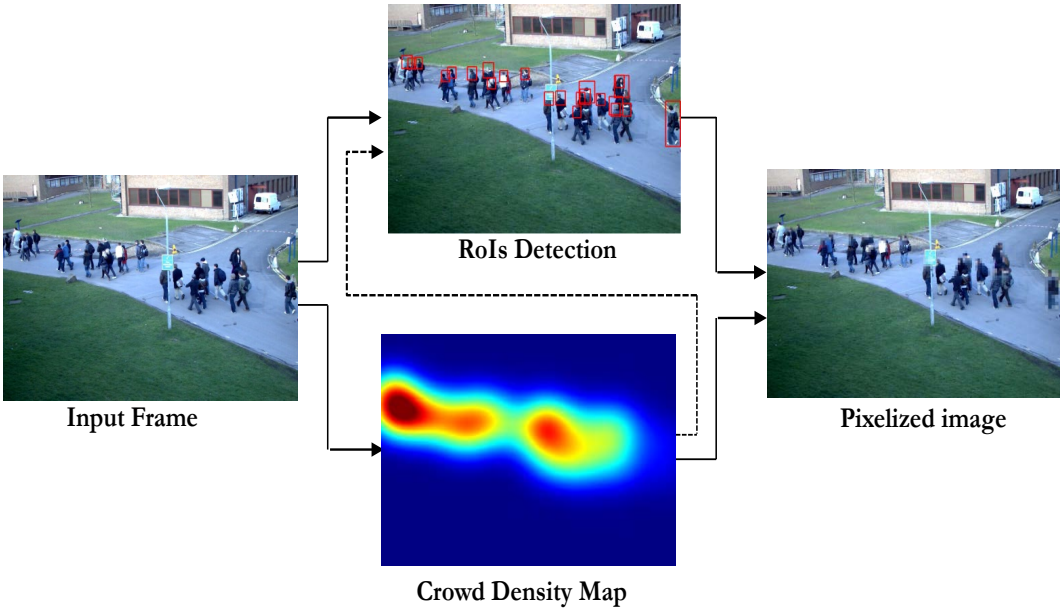


Figure 7.1: Flowchart of the proposed contextualized privacy preservation filters using an examplary frame from PETS 2009 [41], the dotted line in this figure shows that the crowd density map can also be used to improve the robustness of the detection in crowded scenes

7.3.1 RoIs detection

To obfuscate people in the scene, we apply an additional RoI detection step using the deformable part-based models [40]. Firstly proposed in [26], Histograms of Oriented Gra-

dients (HOG) extracts gradient information from a detection window, derives a feature vector from it and compares it against annotated samples. Then, HOG is extended to the deformable part-based models which achieves much more accurate results than the original HOG and marks the state-of-the-art.

As discussed in the previous chapter (Chapter 6), although human detection using part-based models has shown excellent performance, its extension to crowded scenes has limited success. Thus, we follow the same strategy (described in Chapter 6) that consists of using the crowd density map to improve the robustness of the detection in crowded scenes (dotted line in Figure 7.1). As demonstrated before, by integrating the crowd density and geometrical constraints together into the state-of-the-art detector, the detection results are enhanced considerably.

In our framework, we employ this improved deformable part models to detect people in crowded scenes. Then, for people obfuscation, we apply adaptive privacy preservation filters to the head part or to the whole body depending if the target is isolated or within the crowd: If the person is inside the crowd, we limit the obfuscated region to the head, once he is detected as isolated subject (in no-crowded regions), the obfuscation is extended to the whole body. More details about the adaptive protection filters in given in the next paragraph.

7.3.2 Adaptive privacy filters

After applying person detection, we get a set of RoIs $\mathcal{D}_k = \{d_1^k, ..., d_{n_k}^k\}$ at a frame I_k. Given also the crowd density map $C_k(x, y)$ that shows information about the crowd size and the crowd location as well, our goal is to adapt the level of the privacy protection filters according to the crowd density. More precisely, as explained before we intend to apply high privacy protection in less crowded areas while reducing the level of privacy protection in areas with many people. For this purpose, given a set of filter parameters representing different obfuscation levels $P = \{P_{min}, ..., P_{max}\}$, we quantify the crowd density values into $c = |P|$ crowd levels. Then, for a given detection d_j^k, its average crowd density value $\hat{C}_k(d_j^k)$ is used to choose the respective filter parameter that has to be applied to the bounding box d_j^k.

In addition to the crowd density, the visibility of a person in the scene is also sensitive to his distance from the camera because of perspective effects. The perspective distortions can be explained by the fact that persons far away from the camera appear smaller than the closest ones. Thus, the distance from the camera is another parameter that has to be taken into account to choose the suitable obfuscation level. To achieve that, the range of obfuscation levels given by the lower and upper boundary P_{min}/P_{max} is adapted according to the distance from the camera. A simple method to interpret the distance from the camera is to use the size of the detected bounding box. Since this information could be

subject to errors, a better method consists of computing the aspect ratio and the perceived height of a person from all accepted detections (This information can be obtained from the detection step). Using this method, we are able to predict the height $\widetilde{h}_j^k$ and the ratio γ_{k-1} of a detection from the previous detections. Thus, the estimated size of a bounding box d_j^k is $\widetilde{S}_j^k = (\widetilde{h}_j^k)^2 * \gamma_{k-1}$ which is more robust than $w_j^k * h_j^k$.

In this work, we show results for two typical privacy protection filters [32] which are:
Gaussian Blurring:
This privacy filter consists essentially of removing details in a region of interest by applying Gaussian low pass filtering.

$$I_{blur}^k(x,y) = I_k(x,y) * \frac{1}{2\pi {\sigma_{k,j}}^2} e^{\frac{(x^2+y^2)}{2{\sigma_{k,j}}^2}} \tag{7.1}$$

For this technique, the bandwidth $\sigma_{k,j}$ of the used Gaussian is adapted according to the crowd density level and the predicted size.
Pixelization:
This filter is based on decreasing the resolution of any region of interest by replacing each block of pixels in this area with its respective average. The pixelization of frame I_k corresponding to d_j^k detection is given by:

$$I_{pix}^k(x,y) = \frac{1}{b_{k,j}^2} \sum_{i=0}^{b_{k,j}-1} \sum_{j=0}^{b_{k,j}-1} I\left(\left\lfloor \frac{x}{b_{k,j}} \right\rfloor + i, \left\lfloor \frac{y}{b_{k,j}} \right\rfloor + j\right) \tag{7.2}$$

As for the blurring process, the filter size $b_{k,j} \propto (\hat{C}_k(d_j^k), \widetilde{S}_j^k)$.

For both pixelization and Gaussian blurring, the region of interest in restricted to head part only if the person is moving inside the crowd. if $\hat{C}_k(d_j^k) \leq \tau$, then $x \in [x_j^k ... x_j^k + w_j^k - 1]$ and $y \in [y_j^k ... y_j^k + h_j^k - 1]$, otherwise $x \in [x_j^k + \Delta_x ... x_j^k + w_j^k - \Delta_x - 1]$ and $y \in [y_j^k ... y_j^k + h_j^k - \Delta_y - 1]$, where Δ_x and Δ_y parameters are used to crop the head part from the detected bounding box d_j^k.

7.4 Experimental Results

7.4.1 Datasets and Experiments

The proposed framework is evaluated within challenging crowd scenes from multiple video datasets, in particular, some videos from PETS 2009 [41], UCF [4] and Data Driven Crowd Analysis [103] public datasets. To evaluate our proposed context-dependent privacy protection, we adopt an objective evaluation framework, by studying the variation in performances of the state-of-the-art algorithms commonly used in video surveillance analytic before and after applying the proposed privacy protection filters. We recall, as mentioned

in the related works, that one of the major challenges in defining privacy protection policies lies in identifying the correct balance between the two axis of intelligibility and privacy protection of the surveillance data. Therefore, our evaluation framework will consider both axis and model each of them based on the performance scores of an appropriate algorithm.

We model the impact of privacy filters on intelligibility by evaluating the performances of a people counting-by-detection algorithm before and after applying the protection filters. We motivate our choice by observing that privacy protected video surveillance footage must at least retain those visual features necessary to perform very basic monitoring tasks such as people detection and counting.

To evaluate the amount of privacy guaranteed by our method, we model privacy as inverse score of a person matching algorithm based on local features. Such algorithm tries to identify an individual among a set of other subjects by extracting and matching local features between a gallery and a probe set. This algorithm represents a common step for higher level tasks such as person re-identification, recognition or tracking, which could potentially reveal information on the identity of a subject. In our implementation, we use Hessian-Laplace interest point detector together with the SIFT descriptor and nearest neighbor matching, based on the efficient approximate implementation of [89]. Details of the people matching algorithm, together with an extensive evaluation of the different feature extraction and description approaches suitable to the task can be found in [7]. Based on such premise, a good privacy filter should prevent the person matching algorithm to correctly detect and describe local features.

In both cases of intelligibility and privacy, we are only interested in the relative change of performances from the original unprotected images, which constitutes the baseline for privacy filter evaluation. We adopt people counting score as a measure of intelligibility, and one minus person matching score as a measure of privacy protection.

7.4.2 Results and Analysis

In Figure 7.2, the results using three frames from different videos are shown. In the first and the second rows we show the results of RoIs detection, and the estimated crowd density maps. These two sources of information are combined for adaptive protection filters (third and fourth columns). For this purpose, two privacy protection tools (blurring, and pixelization) are employed to show different ways to protect personal privacy in video sequences. In this Figure, it is visible that the block size in the pixelization filter and the bandwidth of the Gaussian blurring are changed by our system according to the crowd density value and perceived size of the person. Comparing e.g. the woman in the lower right corner of the first image row, to the persons walking in the crowd, it is well perceivable that the privacy protection level is reduced within the crowd by a smaller block size or a smaller bandwidth respectively. At the same time, it can be seen that this woman compared to groups of people walking in the crowd does not generate such a high density measure and

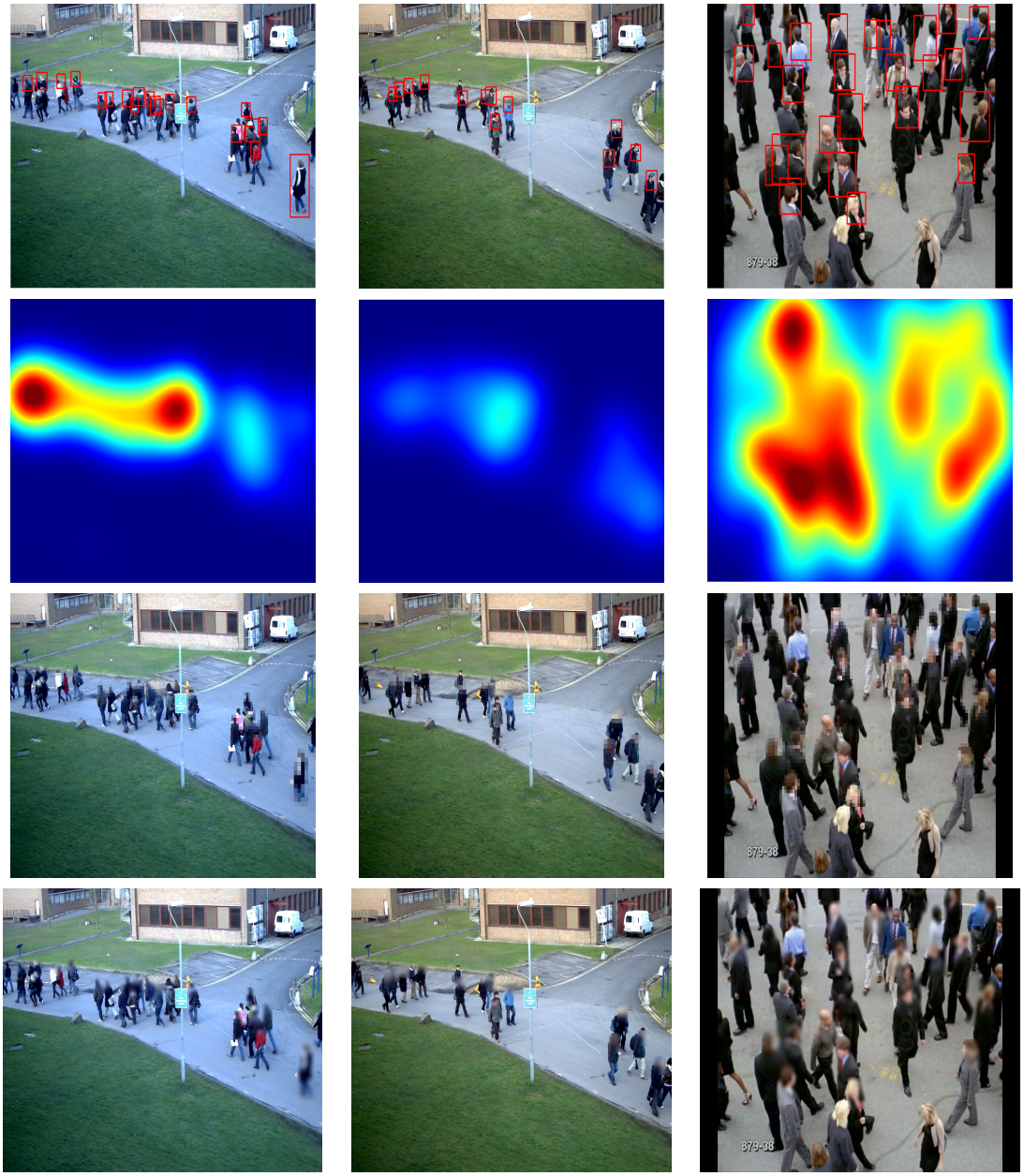
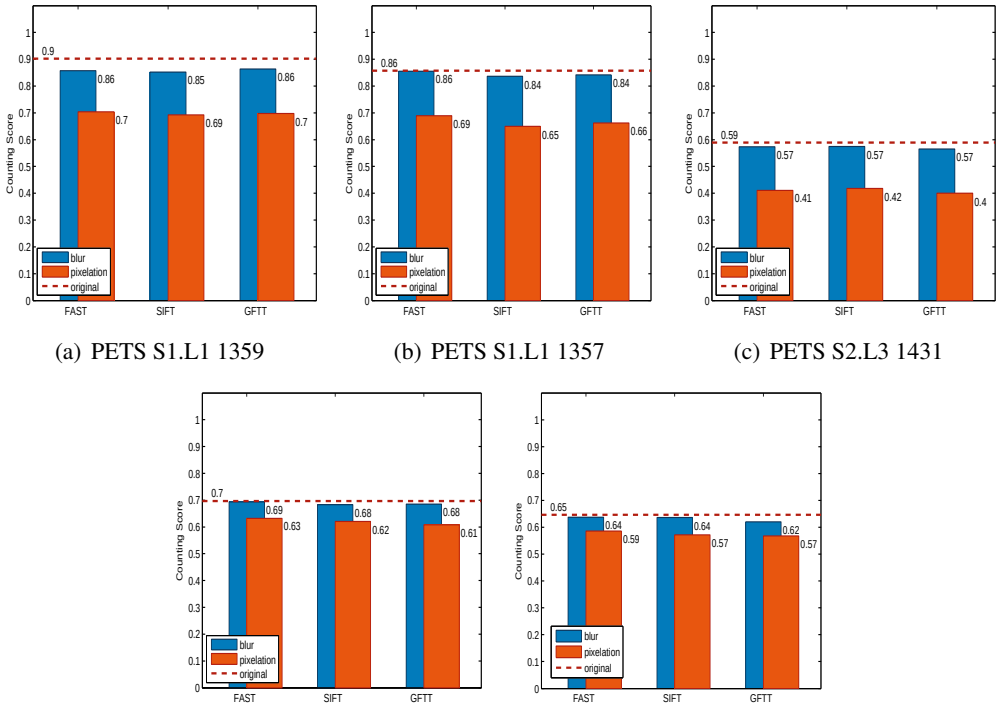


Figure 7.2: Results of adaptive protection filters using three frames from different test videos. From left to right order: PETS2009 S1.L1 1357.V1, PETS2009 S1.L1 1359.V1, and UCF 879. From top to down order: RoIs detection, estimated crowd density map, application of pixelization filter, and application of blurring filter

is consequently obfuscated to a higher degree on the whole body. We also note that the estimated crowd density is lower for the second scene (second column), compared to the first one.That justifies why people in the second scene show rather higher protection levels.

Also, different filter sizes can be seen also using UCF frame (third column).
Visually, the blurring filter seems to be better suited for our application as in general already small block sizes are sufficient in the pixelization filter to render it completely unrecognizable to humans. Nonetheless, our results clearly indicate that crowd density maps are well-suited to improve the crowd context-specific privacy protection in CCTV systems and thus offer a lot of options for further applications.

Following the described evaluation procedure, we test counting and matching on original and privacy protected sequences of PETS, INRIA Data Driven Dataset, and UCF datasets.



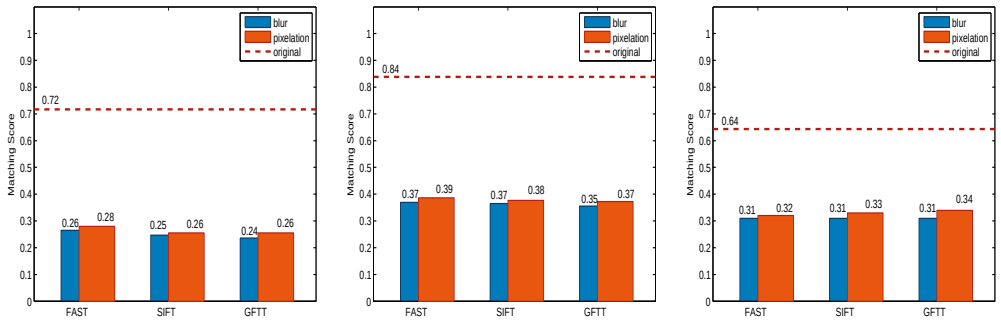
(a) PETS S1.L1 1359 (b) PETS S1.L1 1357 (c) PETS S2.L3 1431

(d) UCF 879 (e) INRIA 879 42

Figure 7.3: Counting scores on sequences protected by blur and pixelization, compared to original results

Figure 7.3 reports the people counting results for blurring and pixelization protection techniques for the different types of features used in crowd density estimation respectively. Since we are interested in evaluating the task of counting people before and after applying a privacy filter, rather than the effectiveness of the counting algorithm itself, a simple evaluation score is chosen, i.e. the percentage $p \in [0, 1]$ of correctly detected individuals with respect to the annotated ones in the ground truth. The red horizontal line represents the counting score when no protection filter has been applied. As a general trend, we can observe that the counting results do not decrease significantly after applying the protection

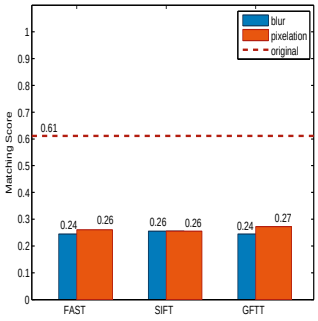
filters. On average, the score drop is 0.10, with 0.03 representing the minimum and 0.18 the maximum loss observed respectively for the blur filter with the SIFT feature and the pixelization filter with the GFT feature. As a consequence, we are still able to correctly perform people counting within a 10% error margin. We also notice that the pixelization algorithm causes the counting to perform worse than the blurring algorithm.



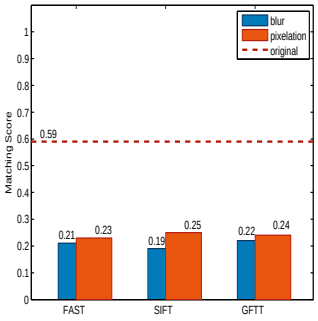
(a) PETS S1.L1 1359

(b) PETS S1.L1 1357

(c) PETS S2.L3 1431



(d) UCF 879



(e) INRIA 879 42

Figure 7.4: Matching scores on sequences protected by blur and pixelization, compared to original results

Matching results are displayed in Figure 7.4, following a convention similar to the previous one for detection. In this case, the red horizontal line represents the baseline matching result when no filter is applied. We can clearly observe a dramatic drop in performances of the person matching algorithm. On average, the drop in matching score is 0.41, with 0.42 and 0.39 being the minimum and maximum observed loss respectively for the pixelization filter with the FAST feature and for the blur filter with the GFT feature.

These results confirm that our approach to privacy protection behaves in accordance to requirements we mentioned in the introduction, in terms of preservation of intelligibility and privacy of the original source. Our privacy protection filters cause a relatively small loss in people counting score, and therefore in intelligibility, compared to the drop in performances of the matching step, and thus the gain in privacy protection level.

We notice as well how in both the counting and the matching experiments, the exact

choice of feature does not influence the results significantly, while it is rather the choice of protection filter which causes variations in the results. Observing that the counting scores, and therefore the intelligibility, is significantly worse in the case of pixelization, and at the same time pixelization offers slightly less privacy protection (higher matching results), the choice of filter falls back on the specific application scenario, according to the desired privacy-intelligibility trade off.

7.5 Conclusion

In this Chapter, we showed how it is possible to include crowd density information into a privacy-preserving framework. Using an additional RoIs detection step, we adapt the degree of data obfuscation for privacy according to the crowd level. By doing so, it is possible to preserve an acceptable level of privacy for the people in a scene while still allowing the operator to view the data relevant for him. As additional contribution, we proposed an objective evaluation of privacy and intelligibility trade-off and we tested that on our contextualized privacy protection filters. By leveraging the state-of-the-art video surveillance analysis algorithms, such as people counting and matching, we demonstrate that our privacy filters retain good performances on common intelligibility tasks such as people counting and detection. At the same time, such privacy filters are able to significantly lower the performances of person matching algorithms based on local features, which potentially can expose identity information of the subject being monitored, therefore threatening their privacy. Furthermore, our evaluation shows that the choice of blur over pixelization as the preferred obfuscation method leads to a better privacy-intelligibility balance.

CHAPTER 8

Crowd Change Detection and Event Recognition

The study of crowd behavior in public areas or during some public events is receiving much attention in security community to detect potential risk and to prevent overcrowd. In this Chapter, we propose a novel approach for change detection and event recognition in human crowds. The proposed approach consists of modeling time-varying dynamics of the crowd using local features tracking which enables removing feature points on the background and extracting long term trajectories. This process is advantageous for the later crowd event detection and recognition since the features irrelevant to the crowd are removed and the tracked features undergo an implicit temporal filtering. As employed in the previous Chapters, these feature tracks are used to generate fully automatic crowd density maps. In addition, they are used to extract regular motion patterns such as speed and flow direction. These attributes (local density, speed, and flow orientation) are modeled by histograms to describe the event or the behavior state of a motion crowd. Finally, crowd change detection is performed by computing their temporal stability, whereas, crowd event recognition is carried out by classifying a feature vector concatenating these histograms.

8.1 Related Works

There are three main categories of crowd behavior analysis methods. The first category is known as microscopic approaches where the crowd is considered as a collection of individuals. To study the crowd behavior; the individuals in the scene need to be segmented, detected and/or tracked. This category includes the Social Force Model [86] which is based on local characteristics of pedestrian motions and interactions, or trajectory-based methods [30, 58]. These methods face considerable difficulties to recognize activities inside the crowd because person detection and tracking tasks are affected by occlusions.

In the second category known as macroscopic methods, the crowd is treated as a whole and a global entity in analysis [17, 12]. These methods are based on extracting the dynamics of the entire scene. For this purpose, scene modeling techniques are used to capture the main features of the crowd behavior. These methods focus on modeling group behaviors instead of determining the motion of individuals, which makes them less complex compared to microscopic methods, thus, they could be applied in analysis of medium to high

crowd dense scenes. Hybrid methods analyze the crowd at microscopic and macroscopic levels. They inherit both properties to handle the limitations of each category of methods and complement each others for better performance [4, 47, 3]. Our proposed method is of hybrid nature as well since it incorporates local optical flow information into extracted local features and it examines long-term trajectories to capture both global and local attributes. This statistical information is further used for high level analysis.

While most of existing works rely on optical flow information between consecutive frames, in our approach we extend this information to build trajectories in order to accurately represent the motion with the video. Also, the generated feature tracks undergo an implicit temporal filtering step which makes them less affected by noise. Another substantial contribution of this Chapter, is the use of local crowd density in addition to the commonly used crowd motion forms (speed and orientation). We consider it as an important cue for early detection of crowd event and it could complement crowd dynamics (motion) information. For example, walking/running events are commonly recognized by measuring the speed which is computed as the mean of the magnitude of motion vectors. However, it is also important to provide additional information about the number or the density of individuals moving at high speed. Other crowd events such as crowd formation have been analyzed using direction of optical flow, again this information is not sufficient, because large number of individuals has to be involved and to participate to crowd formation. Another example that could justify the relevance of using crowd density for event characterization is the blocking situations in large scale crowd, in this case relying only on motion information is not enough since there is no enough spaces to move, as a result the speed slows down. These examples illustrate the need to use density as additional cue for crowd event characterization, also it helps to localize crowded regions.

The remainder of the Chapter is organized as follows: Details about crowd attributes are given in Section 8.2. In Section 8.3, we explain how to use these attributes in order to detect crowd change and to recognize crowd events. A detailed evaluation of our work follows in Section 8.5. Finally, we briefly conclude and give an outlook of possible future works.

8.2 Crowd attributes

The proposed approach is typically based on using local features to represent the individuals in the scene. Also, a feature tracking step is involved in the process of crowd event detection and recognition. By doing so, the daunting task of person detection and tracking is avoided. To achieve an improved overall performance, we consider that density measures could provide rich source of information about the spatial distributions of persons in the scene, mainly for early detection of crowd events such as evacuation, crowd formation, and crowd splitting. Therefore, in our approach we consider simultaneous these both cues of motion vectors: appearance (density) and dynamics (velocity, and direction).

8.2.1 Local crowd density

We propose to use a crowd density measure to complement crowd change detection and event recognition. In particular, we employ the crowd density measure introduced in Chapter 5, which is estimated by measuring how close local features are. Then, a probability density function (pdf) is estimated using a Gaussian kernel density. This process also includes a separation step between foreground and background entities to our system, this can be optimally done using the trajectory of a local feature from the current position to its position in the start frame. Then, static features are identified by comparing its displacements to ζ. The crowd density map is defined as a kernel density estimate based on the positions of the moving local features, for m_k moving local features extracted from a frame I_k at $\{(x_i, y_i), 1 \leq i \leq m_k\}$ positions, the density map C_k is defined as:

$$C_k(x,y) = \frac{1}{\sqrt{2\pi}\sigma} \sum_{i=1}^{m_k} \exp -(\frac{(x-x_i)^2 + (y-y_i)^2}{2\sigma^2}) \tag{8.1}$$

where σ is the bandwidth of the 2D Gaussian kernel.
The estimated crowd density map gives valuable information about the local distribution of people in the scene which is strongly related to the characterization of crowd events.

8.2.2 Crowd motion: Speed and Orientation

The feature tracks are first used to show the spatial distributions of the crowd by estimating crowd density maps based on the positions of moving local features. Second, the same feature tracks are used to extract crowd motion information. It proceeds as follows: after filtering out static features (of zero trajectory lengths because they are stationary along frames, or of small trajectory lengths because of the noise in video acquisition, or dynamic background), for the remaining local features, the overall motion Γ_i^k of a trajectory T_i^k is estimated by computing the euclidean distance between its positions in the start and the current frame. Γ_i^k which denotes the displacement between $(k-\Delta t_i^k)^{th}$ frame to the current frame k is compared to a certain threshold β which is set according to image resolution and camera perspective. The trajectory is considered for further processing only if $\Gamma_i^k > \beta$, while other short-term trajectories of small lengths (occur because of tiny movement of crowd) are filtered out to not affect the computation of speed and orientation. By doing so, the features tracking results are improved and the selected trajectories undergo an implicit temporal filtering which makes them smoother and less affected by noise.

Once the set of useful trajectories is determined, we restrict the history of each 2D trajectory over last few frames because otherwise by considering the whole trajectory an augmentation in the speed will not be detected early and also the flow direction might be less precise. For speed estimation, it is computed as the quotient of the trajectory length divided by the number of frames being tracked. For flow direction, we consider the orien-

tation of motion vector formed by relating the start and the current positions of each local feature.

8.3 Abnormal change detection and event recognition

Overall, the spatio-temporal crowd measures introduced by density maps and motion vectors convey rich information about the distributions and the movements of pedestrians in the scene which are strongly related to their behaviors. For this goal, we first model the crowd attributes by histograms, see paragraph 8.3.1. Then, the application of these attributes for crowd behavior analysis is demonstrated in two steps: First, the variation of a measure of stability (using the histograms) in time is employed to detect change or abnormal event, see paragraph 8.3.2. Second, a feature vector concatenating these histograms is used for event recognition, see paragraph 8.3.3.

8.3.1 Crowd modeling

Each crowd attribute is encoded by 1D-histogram. Given the crowd density map C_k at a frame k, the local density information is quantized into N_d bins. We have chosen $N_d = 5$ according to Polus definition [94] of crowd levels (free, restricted, dense, very dense and jammed flow). Then, to group together motion vectors of the same direction, we quantize the orientation Θ into N_Θ bins. N_Θ is set to 8 bins, which results orientation bin size $\Delta_\Theta = 45$ degrees. As proposed in [29], the speed is quantized into $N_s = 5$ classes: very slow, waking, walking fast, running, and running fast. It is important to note that speed changes can be also affected by perspective distortions, due to the fact that when people are getting away from the camera, their motion vectors are of small lengths. That is why, we rectify these effects on the speed.

8.3.2 Crowd Change Detection

According to the procedure described so far, at each frame k, we obtain three histograms $H_d(k)$, $H_\Theta(k)$, and $H_s(k)$ which denote, respectively, the histograms of density, orientation, and speed. If the motion patterns and the density of the crowd remain similar within a period of time, the corresponding histograms are similar as well. Whereas, if a change occurs in the crowd behavior, that would generate dissimilarities between the histograms.

For histogram comparison in time, we adapt the same strategy as in [29]: we compare the density and the motion patterns at each frame with the those of a set of previous frames. For each histogram $H_i(k)$ at time k, a similarity vector $S_i(k)$ is defined as:

$$S_i(k) = (C(H_i(k), H_i(k - \Delta t_1)), \\ C(H_i(k), H_i(k - \Delta t_2)), ..., C(H_i(k), H_i(k - \Delta t_n))) \tag{8.2}$$

n is the number of frames used in the comparison, Δt_j are the frame steps, and C is the histogram correlation defined between H_1 and H_2 as:

$$C(H_1, H_2) = \frac{\sum_p (H_1(p) - \overline{H_1})(H_2(p) - \overline{H_2})}{\sqrt{\sum_p (H_1(p) - \overline{H_1})^2 \sum_p (H_2(p) - \overline{H_2})^2}} \tag{8.3}$$

where $\overline{H}$ is the mean value of H.
Similar to [29], we define the temporal stability $\sigma_i(k)$ of each histogram $H_i(k)$ as the weighted average of $S_i(k)$:

$$\begin{aligned} \sigma_i(k) &= \omega^T S_i(k), \\ \omega &= \frac{1}{\sum_{j=1}^{n} e^{\lambda \Delta t_j}} (e^{-\lambda \Delta t_1}, e^{-\lambda \Delta t_2}, ..., e^{-\lambda \Delta t_n}) \end{aligned} \tag{8.4}$$

λ denotes the decay constant, $\Delta t_j = j\Delta t$ (Δt is a constant).

In our approach, a change is detected if the similarity between the current frame and the previous frames for one of the crowd attributes (local density, speed, and orientation) is low. For this, we compare each temporal stability $\sigma_i(k), 1 \leq i \leq 3$ to an adaptive threshold $\tau_i(k)$ computed as the half average of the temporal stability values σ_i between $(k - \Delta t_1)$ and $(k - \Delta t_n)$:

$$\tau_i(k) = \frac{1}{2n} \sum_{j=1}^{n} \sigma_i(k - \Delta t_j) \tag{8.5}$$

8.3.3 Event Recognition

The proposed crowd attributes are also used to recognize crowd events. In particular, 6 crowd events are modeled namely, walking, running, evacuation, local dispersion, crowd formation and crowd splitting. In our approach, we propose to perform event recognition by classification. For testing, given a new frame $\mathbf{x}$, we aim at classifying it into one of the events $y^* \in \mathcal{Y}$, which maximizes the conditional probability:

$$y^* = \arg\max_{y \in \mathcal{Y}} P(y|\mathbf{x}, \theta^*) \tag{8.6}$$

where θ^* are learned from the training data. This can be performed by SVM classification, and for the feature vector, we concatenate the 3 histograms $H_d(k)$, $H_\Theta(k)$, and $H_s(k)$ into $\mathcal{H}_k$. For classification, we use Chi-Square kernel:

$$K(\mathcal{H}_i, \mathcal{H}_j) = \sum_I \frac{\mathcal{H}_i(I) - \mathcal{H}_j(I))^2}{\mathcal{H}_i(I) + \mathcal{H}_j(I)} \tag{8.7}$$

8.4 Crowd event characterization

We consider that the local density is an important cue to characterize crowd events. In addition, it provides helpful information about the density of people that participate to a detected event, also it is useful to localize the event since it is estimated at local level. The characterization of crowd events is as follows:

8.4.1 Walking/Running:

Walking event corresponds to a number of persons moving at low speed. If the speed is high, running event is detected. This can be recognized by computing the mean of magnitudes of all motion vectors at each frame. Obviously, an abnormal event occurs when there is change from walking to running. Also, more attention should receive running event when large number of persons is involved, that could be given by the estimated crowd density which enables the localization of the event as well.

8.4.2 Evacuation:

Evacuation is defined as a sudden dispersion of the crowd in different directions. To recognize this event, direction, speed, and crowd density attributes can be used. This event can be characterized by detecting more than 4 principal directions which have to be distant from each others. Also, a degradation in the crowd density and an increase in the speed and in the motion area have to be detected to recognize this event.

8.4.3 Crowd Formation/Splitting:

Crowd formation (or merging) event is recognized when we detect a merge of many individuals coming from different directions towards the same location. For this purpose, distance between main directions can be used. Also, this event is characterized by an increase in the crowd density and a decrease in the motion area. The opposite of crowd formation is crowd splitting event.

8.4.4 Local Dispersion

This event is recognized when people moves locally away from a threat. The same attributes of crowd formation and splitting can be used.

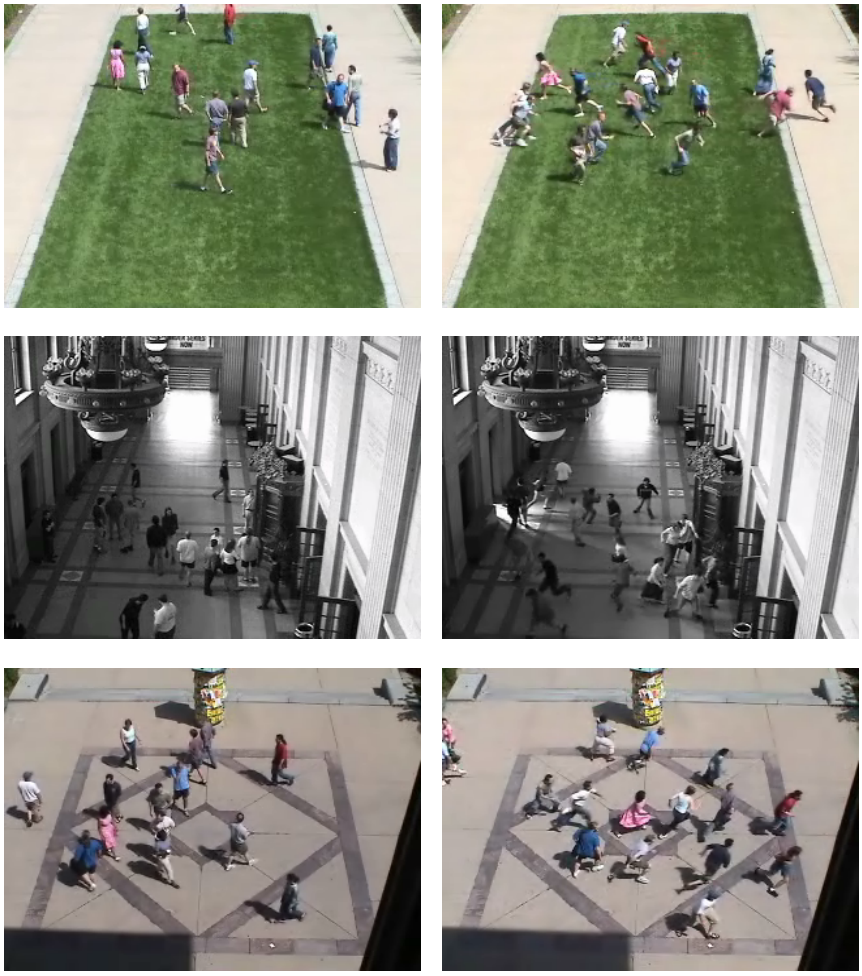


Figure 8.1: Sample frames for UMN dataset. From top to bottom: Scene 1, Scene 2, and Scene 3. From Left to Right: samples of normal and abnormal events from each scene.

8.5 Experimental Results

8.5.1 Datasets

To evaluate our proposed approach for crowd change detection and event recognition, we use two public datasets: PETS 2009-S3 dataset and the dataset of the University of Minnesota (UMN) [1]. The Section S3. Event Recognition of PETS 2009 dataset has been employed to assess crowd event detection and recognition. The public UMN dataset has been widely used to distinguish between normal and abnormal crowd activities.

First, for crowd change detection, we test our proposed approach on the publicly available UMN dataset. The dataset comprises 11 videos in three indoor and outdoor scenes organized as follows: Videos 1-2 belong to scene 1, Videos 3-8 belong to scene 2, and the

scene 3 consists of Videos 9-11. Figure 8.1 illustrates some samples of these three scenes. Each of these videos can be divided into normal and abnormal parts. Precisely, they illustrate different scenarios of escape event such as crowds running in one direction, or people dispersing from a central point.

For the ground truth, as noticed in some previous works [29, 19], the labels of abnormal events shown in the videos are not accurate. There are some time lags in the ground truth labels, for instance in Video1, according to the labels of the ground truth, it is shown that an abnormal event occurs from frame 526, however people started running at frame 484. To overcome this conflict, we use the labels of change detection of some videos provided in [29, 19], for the other videos we follow the same annotation strategy; we manually label the frame in which the crowd change happens (in particular, in UMN dataset as soon as people start running).

For evaluating crowd event recognition, we test our method on PETS 2009. S3, used to assess crowd event recognition algorithms. This dataset comprises 4 video sequences with the following time-stamps 14:16, 14-27, 14-31 and 14:33 and only one view is used for our experiments (View1). As noticed in [47], some sequences are composed of 2 video clips, this is the case of 14:16, 14:27, and 14:33, which results 7 videos in general. More details about these 7 videos are given in Table 8.1.

sequence name	first frame	last frame
14:16-a	0	107
14:16-b	108	222
14:27-a	0	184
14:27-b	185	333
14:33-a	0	310
14:33-b	311	377
14:31	0	130

Table 8.1: Videos from PETS2009. S3 used for testing crowd events recognition algorithms: the first and the last frames of each video sequence.

These videos depict 6 classes of crowd events: walking, running, formation (merging), splitting, evacuation, and dispersion. We annotate these videos with the 6 classes as it is shown in the following Table 8.2.

8.5.2 Experiments and Analysis

8.5.2.1 Crowd Change detection

For evaluating crowd change detections, accurate detection means early detection as soon as the change occurs. For quantitive evaluation, we employ the relative mean frame error

events	video [frames]
walking	seq.14:16-a [0-40], seq.14:16-b [0-56]
running	seq.14:16-a [41-107], seq.14:16-b [57-114]
evacuation	seq.14:33-b [24:66]
dispersion	seq.14:27-a [96:144], seq.14:27-b [86:134]
formation	seq.14:33-a [0:180]
splitting	seq.14:31 [58:130]

Table 8.2: The time intervals indicate where a specific event is recognized (from its first frame to the last one)

metric proposed in [64]. It is defined as:

$$e_F = N_e/N_{fr} \tag{8.8}$$

where N_{fr}, N_e denote the total number of frames in the video, and the error frames, respectively, see Table 8.3.

Seq. UMN	Nb Frames	Ground Truth	Our Det. changes	e_F
Video1	625	484	493	0.0144
Video2	828	665	669	0.0048
Video3	549	303	319	0.0291
Video4	685	563	582	0.0277
Video5	769	492	512	0.0260
Video6	579	450	466	0.0276
Video7	895	734	754	0.0223
Video8	667	454	471	0.0255
Video9	658	551	551	0
Video10	677	570	577	0.0103
Video11	807	717	722	0.0062

Table 8.3: Comparison of our detection results to the ground truth labels using error frame metric

As demonstrated in Table 8.3, the comparison of our detection results to the ground truth labels shows satisfactory performance and rather accurate in most videos. The delay in the detection is more visible in the second scene (from Video 3 to Video 8). In terms of e_F metric (the last column in the Table), the error is small in most cases. In our approach, the delay in the detection of some frames after the event occurs is because of our strategy of detection, in which an abnormal event is detected if the temporal stability is below the

dynamic threshold (defined as half the average of temporal stabilities of previous frames). This requires some times to be detected, which justifies the delay. At the same time, this strategy is suitable to avoid false alarms (accuracy/precision trade-off).

To demonstrate the effectiveness of our proposed approach, we compare our results to other methods, namely, the Social Force Model (SFM) [86], the adjacency-matrix based clustering (AMC) [19], and the similarity metric based on 2D-histograms decoupling speed and orientation in [29]. Figures 8.2, and 8.3 illustrate our results compared to these methods on some videos of UMN dataset. In these figures, the green bar indicates normal events, and the red color denotes the abnormal event detected or labeled (in the ground truth). These results show that our method gives better results than SFM and comparable results regarding the two other methods. It is important to mention that UMN does not include events such as crowd formation/splitting, that could justifies how the methods based only on motion information (speed and orientation) could achieve satisfactory results. More tests on crowd events are required to demonstrate the usefulness of local crowd density as additional attribute for crowd event detection and recognition.

Furthermore, precision and recall of our proposed approach are listed in Table 8.4. We compare our results to (AMC) method [19], which also runs on the same dataset and labeled the ground truth manually. In fact, the conflict concerning the ground truth annotations impeded additional comparisons. This comparison shows that our method achieves comparable results in terms of recall. 100% is achieved in terms of precision which means zero false alarms for all videos, however, the evaluation in terms of precision is not provided for the compared method [19]. For recall (or detection rate) we get worse results, but of small margin, for the same reason mentioned before about time lags in the detection until the similarity metric becomes less than the dynamic threshold.

Approach	Recall (%)	Precision (%)
Proposed approach	92.45	100
AMC approach	94	n/a

Table 8.4: Performance of our proposed crowd change detection method in terms of recall and precision using UMN dataset compared to [19]

8.5.2.2 Crowd Event Recognition

For crowd event recognition, we randomly split the dataset PETS 2009. S3 into (75%) for training and (25%) for testing. This random split is done 10 times, and the following results are the average of these 10 iterations. For each test sample, the feature vector using the concatenation of the three histograms is identified as one of the six classes following one-vs-one strategy. We obtain (99.54%) as classification accuracy. We also evaluate the recognition performance with confusion matrix, see Table 8.5.

	walking	running	splitting	dispersion	evacuation	formation
walking	0.9958	0.0042	0	0	0	0
running	0.0032	0.9968	0	0	0	0
splitting	0	0	1.0000	0	0	0
dispersion	0	0	0	1.0000	0	0
evacuation	0	0	0	0	0.9794	0.0206
formation	0	0	0	0.0067	0	0.9933

Table 8.5: Confusion matrix for event recognition on PETS 2009. S3 dataset

In addition, we report the classification accuracy on the test set for each class separately, following one-vs-rest strategy, see Table 8.6.

Events	Walking	Running	Splitting	Dispersion	Evacuation	Formation
accuracy	99.41	99.21	100.00	99.87	99.80	99.54

Table 8.6: Classification accuracy of our proposed crowd event recognition method on test set from PETS. S3 dataset following one-vs-rest strategy

As it is shown in these tables 8.5, 8.6, we achieve excellent results for all crowd events including crowd formation/splitting, which justifies the relevance of our proposed attributes.

8.5.2.3 Crowd Characterization

For evaluating our proposed crowd event characterization, we use PETS 2009. S3 dataset. By following up some measures extracted from the crowd attributes (unsupervised method), we achieve better video understanding; precisely, we are able to monitor the variation of crowd attributes in time, to interpret what is happening in the scene, to localize the event, and to have clear idea about the density of people participating to each event. Figure 8.4 illustrates some examples of event characterization on PETS 2009.

In the first row of this figure, we show a sample frame of crowd formation. This event is characterized by people coming from different directions and they are moving towards the same location (as it is depicted in the first column, showing the direction of motion vectors). Also, this event is characterized by a decrease of motion area ratio in time, in this frame it is equal to 40.72%. In the second column, we show the estimated density map, which localizes where the crowd is formed. The area of dense regions is augmenting in time, it reaches 6.10% at this frame. Given all the characteristics, crowd formation event is recognized and localized as it is shown in the third column.

In the second row, we show an example of evacuation. This event is characterized by the

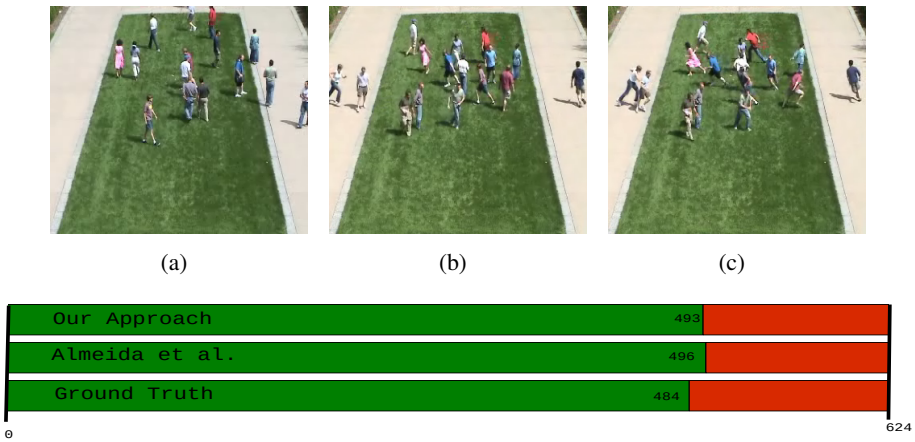


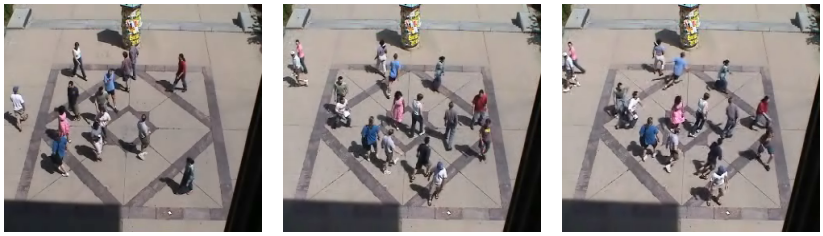
Figure 8.2: Results on Video1 of UMN [1] dataset (a) The first frame of the video sequence (b) The frame in which the crowd change occurs (c) The frame in which our method detects the crowd change (d) Comparisons of our result to [29] result and to the ground truth

divergence of motion vectors as it is shown in the first column, because people are moving away from each others in different directions. In addition this event is characterized by a sudden increase in the speed; the average of magnitude of all motion vectors at this frame is equal to 12.48 pixels (the effects of perspective distortions are considered in the computation). This event is also characterized by in an increase in the motion area ratio (53.79%) and a decrease in time of dense areas (as it is shown in the second column).

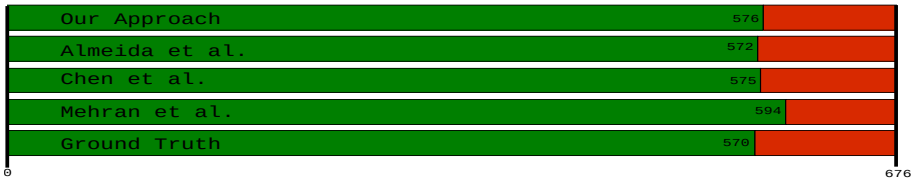
8.6 Conclusion

In this Chapter, we proposed a novel approach to automatically detect abnormal crowd change and to recognize crowd events in video sequences based on analyzing some attributes of crowd tracks. In addition to the increasing need for automatic detection and characterization of crowd events, our study is motivated by the necessity of implying density estimation in the process because the risk of dangerous events increases when a large number of persons is involved. The effectiveness of using local density together with motion information has been experimentally validated using videos from different crowd datasets. The results show good performance for early detection of crowd change and accurate event recognition.

Because crowd events have temporal structure, Hidden Markov Models (HMM) can tackle this classification better than SVM (classification per-frame which disregards temporal order) by capturing temporal patterns in the data. The small size of PETS 2009.S3 dataset impeded us to investigate this method, since HMM requires extensive training data. Another future direction of this work could be the use of the same input (local features



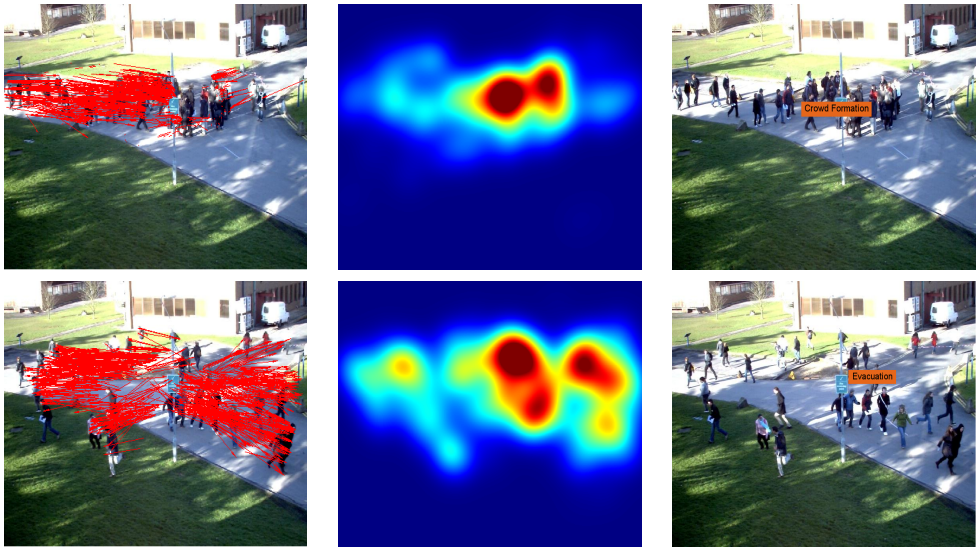
(a) (b) (c)



(d)

Figure 8.3: Results on Video10 of UMN [1] dataset (a) The first frame of the video sequence (b) The frame in which the crowd change occurs (c) The frame in which our method detects the crowd change (d) Comparisons of our result to [29, 19, 86] results and to the ground truth

tracking) to study group behaviors by performing trajectory clustering.



a) Motion Vectors (b) Density map (c) Recognized event

Figure 8.4: Results of event characterization from PETS 2009 dataset.

CHAPTER 9

Conclusions and Future Perspectives

In this thesis, new insights into crowd density analysis are highlighted and studied. Specifically, our contribution to crowd analysis field covered different aspects including: improving crowd density estimation compared to the baseline methods, extending this estimation from global level to pixel level using sparse feature tracking, enhancing human detection and tracking in crowded scenes using a prior estimation of local density, applying the crowd density in privacy context to boost the compliance between surveillance and privacy concerns, and capturing crowd behaviors that occurs over spatio-temporal extents.
This Chapter summarizes these contributions concludes by describing ongoing and future research directions of this work.

9.1 Conclusions

In the first part of the thesis, we addressed problems related to crowd density estimation and characterization. Our proposed approaches treated local features as the main cues instead of individuals to avert typical problems encountered while applying detection and tracking in crowded scenes.

In Chapter 3, we addressed the people counting problem, which is a crucial component in the field of crowd density analysis. We highlighted our contribution through two different proposed approaches. Some enhancements were achieved in this field either in the formulation of the problem or in the obtained results compared to other regression-based methods. In addition to the problem of perspective distortions which is widely addressed in the literature, we handled the problem of crowd density variations in a slightly different way by formulating a new weight function based on local features density for crowd normalization. Our proposed methods consisted of regressing a single frame-wise feature independent from variations of perspective and crowd density. The experiments demonstrated that our approaches achieve good counting accuracy in situations of important occlusions and perspectives distortions compared to other existing methods.

In Chapter 4, we addressed crowd level estimation which is an alternative representation of crowd density. In particular, we focused on texture analysis to characterize the crowd density at patch level. Then, we applied PCA and LDA to enhance the discriminative and descriptive power of LBP features. Furthermore, we included a large comparative study to prove that among numerous texture features only few of them are discriminative to

the crowd. We also proposed a new multiclass SVM algorithm using automatic relevance scores. The experimental results highlighted the role of low-dimensional compact representation of LBP on the classification accuracy and demonstrated that our proposed feature vector is robust enough to perform well in different levels of the crowd. Furthermore, the application of the proposed multi-class algorithm demonstrated good results in terms of classification accuracy while maintaining lower computational cost over other existing multiclass SVM methods.

In Chapter 5, we extended crowd density estimation to local level via kernel density estimate based on the positions of moving local features. By this way, local information of crowd density was introduced at pixel level instead of providing global information per frame or at patch level. The crowd density information was represented as a new statistical model of spatio-temporal local features that varies temporally over the video and spatially across the frame. This process included a feature tracking step to alleviate the effects of features irrelevant to the crowd density. Our proposed approach was tested on videos from different datasets. Following our proposed evaluation methodology, the results demonstrated the effectiveness of sparse feature tracks for crowd estimation. Furthermore, we included a comparative study between different local features in order to investigate their discriminative power to the crowd.

Along the first part of the thesis, our contributions were described within different video analysis components. An overview of these components can be categorized into three main classes: (1) Visual features extraction: it introduces the various methods that we used to transform the raw data of a video sequence to a more sophisticated description. This description illustrates different properties of an object or an event regarding the tackled problem. It varies from dense to sparse, and from local to global features. (2) Motion estimation: this is a fundamental step in our work, since our study is typically focused on crowd dynamics. Along the first part of the thesis, we applied different techniques, namely, background subtraction using GMM, dense optical flow by Farneback algorithm, and sparse optical flow by RLOF. (3) Pattern recognition and machine learning: using machine learning and pattern recognition techniques, the set of extracted features are classified, clustered, and regressed depending on the task we want to perform. Precisely, we applied classification by Support Vector Machine, regression by Gaussian Process, clustering by DBSCAN technique, and dimensionality reduction by PCA and LDA.

In the second part of this thesis, we approached some problems related to the crowd analysis field from a new perspective. Given the difficulties encountered by video analytic components in crowded scenes, we used a prior estimation of crowd density to complement these components in failure cases in large scale crowds. In particular, we employed the proposed local space-time model of crowd density introduced in Chapter 5 for the following purposes:

Since human detection and tracking are challenging in crowded scenes, in Chapter 6 we introduced additional knowledge about the spatial distribution of persons in the scene us-

ing the local density map in the detection process. The combination of crowd density with the localization of individuals was performed through the use of scene-adaptive dynamic parametrization. The improved detection results were extended to tracking in a tracking-by-detection framework. The evaluation of our method on videos from different datasets demonstrated that our system achieves better detection results than the baseline algorithm (the state-of-the-art detector). This resulted in substantial improvements for tracking results since such paradigm of tracking involves the continuous application of detection in individual frames and the association of the detection across frames.

Second, we showed in Chapter 7 that density estimates could be used to adjust the privacy level. By means of automatically estimated crowd density maps, the obfuscation level decreases according to the local density. Our work in this field leaded to proposing new contextualized privacy protection filters, which are effective in high density scenes as well as in low density scenes. In addition, an objective evaluation was proposed to assess intelligibility vs. privacy. The experiments results demonstrated that our proposed context-aware privacy filters give good performance on common intelligibility tasks (such as people detection and counting), while protecting the privacy of persons being monitoring (difficulties to match people).

Finally, Chapter 8 was dedicated to crowd change detection, and crowd event recognition. Our proposed approach employed low-level features in favor of high-level applications and bypassed mid-level features like object detects and tracks. The crowd tracking was based on long-term trajectories of local features. Then, the crowd change detection and crowd event recognition were performed by modeling local density and motion patterns extracted of these long-term trajectories. The experimental results demonstrated good performance for early detection of crowd change and accurate crowd event recognition.

To sum up, in the second part of the thesis, we have demonstrated that there are several advantages of estimating crowd density for use in practice. Although estimating crowd density from videos is an important visual surveillance task for crowd management and monitoring purposes to detect overcrowding situations, it could be used to complement other applications in video surveillance. Specifically, three different applications were investigated.

9.2 Limitations, extensions and directions for future research

There are several possible extensions of this work:

- In Chapter 3, our study of people counting has been validated only on medium crowded scenes. No extremely dense scene were used for tests. To be able to achieve that, fusion of different of features could be used as recently proposed in [57].

- In Chapter 4, it be could interesting to improve the results of crowd level classification by adding other features. In fact, texture features are relevant to crowd density

estimation, but in some cases, these features are not efficient enough (e.g. in case of low crowd density and complex texture because of colorful clothes or in case of high crowd density and simple texture). These cases lead to misclassification, which could be handled by using other features.

- In Chapter 4, although the use of PETS 2009 for crowd level classification is relevant since it is a well known dataset in video surveillance community, we had some difficulties to extract enough samples for training and testing sets, also jammed crowd level could not be investigated using this dataset. Therefore, using more challenging datasets for tests could be planned as a perspective.

- In Chapter 5, for our proposed crowd density map method, we assume that the cameras are static. Actually, the feature tracking step could operate on moving/PTZ cameras (in contrast to the often-used background subtraction), however, as the density estimation relies on separation between moving/static features, it will suffer from the respective performance loss. By adding a global motion estimation step from the features, this problem could be alleviated.

- In this thesis, our study is mainly based on capturing the dynamics of crowd, in particular, we assume that only persons are moving in the scene. Since this assumption is not always true, it could generate false detections if other objects are moving in the scene or persons are not moving. Therefore, object categorization could be included in the process to distinguish between persons and other objects in the scene.

- In Chapter 6, we formulated an improved person detection using crowd density estimates. Then, the improved detections are extended to tracking by employing tracking-by-detection. A more elegant approach could formulate both detection and tracking as a joint framework and crowd density information could be integrated in both steps to enforce scene constraints. One way to do that is to use the density information in the likelihood function.

- In Chapters 6 and 8, additional and suitable priors (in addition to crowd density measure) for improving human detection and tracking in crowded scenes, and for recognizing crowd events are acceptable.

- In Chapter 7, only objective evaluation is used to assess our proposed contextualized protection filers; it could be interesting to perform subjective evaluation as well.

- In Chapter 8, we can detect the crowd events since they start, or after they occur, however, it could be interesting to investigate how to predict an event before it happens.

- One of the difficulties that we encountered in this work, is the heavy hand-annotation task. All available datasets for crowd analysis field, are not annotated and because

of the complexity of crowded scenes (small size of objects and interactions between them), annotating them remains a challenging task, if not impossible in some cases. Therefore, we believe that more progress can be achieved in this field if much effort can be devoted by providing annotated datasets.

- Given the difficulties encountered in the analysis of crowded scenes, we believe that using new sensors could overcome many limitations. These sensors could be range sensors; by providing additional depth information, some enhancements in the performance of video analysis are expected, for example, perspective distortions problem would be automatically addressed. However, it is important to note that, so far the available RGB-D sensors have limited depth range, which could prevent the use of these sensors in outdoor applications. Likewise, new perspectives in crowd analysis field are expected using moving cameras (such as Unmanned Aerial Vehicles, wearable cameras for police officers, and car-mounted cameras). For instance, a quadcopter-mounted camera could follow a crowd, however static surveillance systems could not reach it.

APPENDIX A

Foreground Segmentation

A.1 Introduction

Foreground segmentation consists of separating the moving objects from the static part of the scene. Among the proposed methods to handle this problem, GMM has shown substantial improvement by adopting more variety in the background. It is based on a probabilistic approach that achieves satisfactory performance thanks to its ability to handle complex background scenes. However, the background/foreground discrimination still leaves rooms for further improvements. Actually, the decision on which of the Gaussians are most likely belonging to the background is made based on selecting the ones having the most supporting evidence and the least variance, which is not always correct. This ambiguity left by GMM method makes the estimation of the background model still hard to be properly addressed. To overcome the mentioned shortcoming and to achieve an improved overall performance, we propose to incorporate an uniform motion model into GMM background subtraction. Considering these both information over time into a single overall system has the potential to detect foreground objects more reliably.

A.2 Baseline Method: Background subtraction by Gaussian Mixture Model

The most popular background subtraction algorithm is based on Gaussian mixture model proposed by Stauffer and Grimson [120]. This method uses a mixture of K Gaussian distributions to model the recent history $\{X_1, ..., X_t\}$ of each pixel. The probability of observing the current pixel value is defined by a sum of weighted Gaussian distributions :

$$P(X_t) = \sum_{i=1}^{K} w_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \tag{A.1}$$

where K is the number of distributions (typically between 3 and 5), $w_{i,t}$, $\mu_{i,t}$ and $\Sigma_{i,t}$ are respectively, an estimate of the weight, the mean value and the covariance matrix of the i^{th} Gaussian in the mixture at time t. And $\eta(X_t, \mu, \Sigma)$ is the Gaussian probability density function. Then, incoming pixels are compared against the corresponding Gaussian mixture model in order to find a Gaussian within 2.5 standard deviations. If a matching is found,

the mean and the variance of the matched Gaussian are updated accordingly. However, if there is no match, the least probable distribution is replaced by a new one modeling the incoming pixel. The prior weights of the K distributions at time t are defined as follows:

$$w_{k,t} = (1 - \alpha)w_{k,t-1} + \alpha(M_{k,t}) \tag{A.2}$$

where α is a learning rate, and $M_{k,t}$ is equal to 1 for the matched model and equal to 0 for the remaining models. The updated parameters of the distribution that matches the new observation are defined by:

$$\sigma_t^2 = (1 - \rho)\sigma_{t-1}^2 + \rho(X_t - \mu_t)^T(X_t - \mu_t) \tag{A.3}$$

$$\mu_t = (1 - \rho)\mu_{t-1} + \rho X_t \tag{A.4}$$

where σ_{t-1}, μ_{t-1} are the last mean and the variance values of the matched Gaussian and X_t is the value of the new pixel. ρ is the second learning rate defined by:

$$\rho = \alpha(X_t|\mu_k, \sigma_k) \tag{A.5}$$

The last step aims at estimating the background model from the mixture. For this purpose, the algorithm assumes that Gaussian distributions having the most supporting evidence and the least variance are most likely produced by background processes. As a result, the Gaussians are ordered by w/σ and the first B of the ranked distributions whose accumulated weights exceed T are deemed to be the background:

$$B = \arg\min_b (\sum_{k=1}^{b} w_k > T) \tag{A.6}$$

where T is the minimum fraction of the background model.

The detailed adaptive modeling background is robust enough for illumination changes; it can also deal with the movement in the background due to its multimodality. Many improvements for this method can be found in the literature to solve different limitations. One of these limitations arises due to the use of fixed learning rate all the time. Therefore, the parameters stabilize slowly which leads to problems with the initialization. That is why, the original version of GMM background subtraction has been further enhanced to improve its learning rate. This modification was proposed by Kaewtrakulpong and Bowden [62]. For the initialization, they improved the slow learning problem by using online Expectation Maximization algorithm and switching to the L-recent window update equations in order to give priority over recent data. This makes the convergence on a stable background model faster and also the tracker adapted to changes in the environment.

Another limitation of GMM method is caused by using a fixed number K of components over the time. Also, this number remains the same for each pixel which is not opti-

mal in terms of computational time and segmentation accuracy. To address this problem, Zivkovic [138] proposes to constantly update not only the parameters but also the number of components of the mixture for each pixel. Using the Dirichlet prior, an online algorithm estimates the parameters of the GMM and selects the appropriate number of Gaussians simultaneously. As a result, K is dynamically adapted to the multimodality of each pixel. This method is called improved adaptive Gaussian mixture model and it is developed with shadow detection [95] to remove moving shadow pixels upon pixels labeled as foreground. A pixel is detected as shadow if it is considered as darker version of the background. For this purpose, a threshold is used to precise how much darker the shadow can be.

Even if these modifications proposed in [138] showed improvement comparing to the original algorithm, the separation between foreground and background distributions is still problematic. Actually, the distinction is based on selecting as background components the Gaussians that are more frequently matched. In other words, it assumes that the often occurring pixels are deemed to model the background, which it is not always the case. That is why; we propose combining the improved GMM background subtraction with a uniform motion model into a single framework. This observation leads to better segmentation of the scene into foreground and background entities.

A.3 Improved Foreground Segmentation Using Uniform motion estimation

As explained before, the estimation of the background model using GMM still leave some ambiguities. At the same time, motion cue can provide additional and important information about the scene structure. That is why, we propose combining motion information with GMM background subtraction. This combination of both cues is based on the assumption that pixels moving together (with the same velocity and orientation of the optical flow) have to get the same label (foreground or background). For this purpose, a measure for uniformity of motion is defined. Then, the incorporation of these two cues is done by favoring similar labels for pixels moving together.

The second cue of the proposed approach is motion information. It is obtained by computing the optical flow between consecutive frames, then a measure for uniformity of motion is applied. For optical flow computation, we apply the method proposed by Farneback [39], it consists of computing optical flow based on polynomial expansion. This method uses quadratic polynomial model to approximate each neighborhood of both frames. Then, it estimates displacement fields from the polynomial expansion coefficients by observing how an exact polynomial transforms under translation. Also, it uses Gaussian to smooth the neighboring displacements. The evaluation of this method shows good results in terms of accuracy and low computation burden.

Figure A.1 shows the optical flow across two adjacent frames at times t and $t + 1$. For

(a) *First frame* (b) *Second frame* (c) *Optical flow*

Figure A.1: *Dense optical flow computation for two consecutive frames*

each point P located at the 2D image coordinate $\vec{x} = [x\ y]^T$, the dense optical flow field provides a motion vector which is expressed as $2D$ velocities $\vec{V} = [v_x\ v_y]^T$. From these $x-$ and $y-$ components of the 2D velocity field, the optical flow filed of each point P in the origin image can be also defined by its magnitude and its direction as follows:

$$Optical\ Flow(P_{x,y,t}) = \begin{pmatrix} Magnitude\ (P_{x,y,t}) \\ Direction\ (P_{x,y,t}) \end{pmatrix} \tag{A.7}$$

The magnitude of motion is convoluted with the difference between each current frame and the mean of the background to get precise boundaries. After that, the detection of uniform motion is performed. It works as follows: only pixels having non-zero optical flow velocities are considered. For the remaining values in the magnitude of the motion, neighbor pixels having similar direction are grouped in the same component. Therefore, four directions corresponding to these quadrants {[$-\pi/2$, 0], [0, $\pi/2$],[$\pi/2$, π],[$-\pi$, $-\pi/2$]} can be distinguished. For each direction d, N_d connected components are obtained with different brightness values for the magnitude. To measure the uniformity of the motion inside each component, a mean motion value is computed. If we denote Ω_k one component of the current frame, where k varies from 1 to N (N is the total number of components expressed as: $N = \sum_{d=1}^{4} N_d$), the mean motion value inside Ω_k is defined as follow:

$$v_k = 1/p \sum_{i \in \Omega_k} v_i \tag{A.8}$$

where p is the total number of pixels inside Ω_k and v_i is the magnitude of the motion for a pixel i. The difference between each magnitude value and the mean value inside the component is used as an error measure:

$$\varepsilon = v_i - v_k \tag{A.9}$$

Then, an adequate threshold is chosen empirically for measuring the uniformity of motion.

Only pixels belonging to Ω_k and regarding this uniformity will be considered. After the distinction between the different new components Ω'_k (with the same velocity and orientation), the label of each component Ω'_k (whether it belongs to background BG or foreground FG process) is defined as follows:

$$label(\Omega_k^{'}) = \begin{cases} FG \text{ if } (\frac{\sum_{(\forall P_i \in \Omega'_k, P=1)} E(P_i,P)}{M_k} \geq R) \\ BG \text{ otherwise} \end{cases} \tag{A.10}$$

where M_k is the total number of pixels inside Ω'_k, R is a ratio in the range of $[0, 1]$,

$$E(P_i, P) = \begin{cases} 0 & \text{if } label(P_i) = label(P) \\ 1 & \text{if } label(P_i) \neq label(P) \end{cases} \tag{A.11}$$

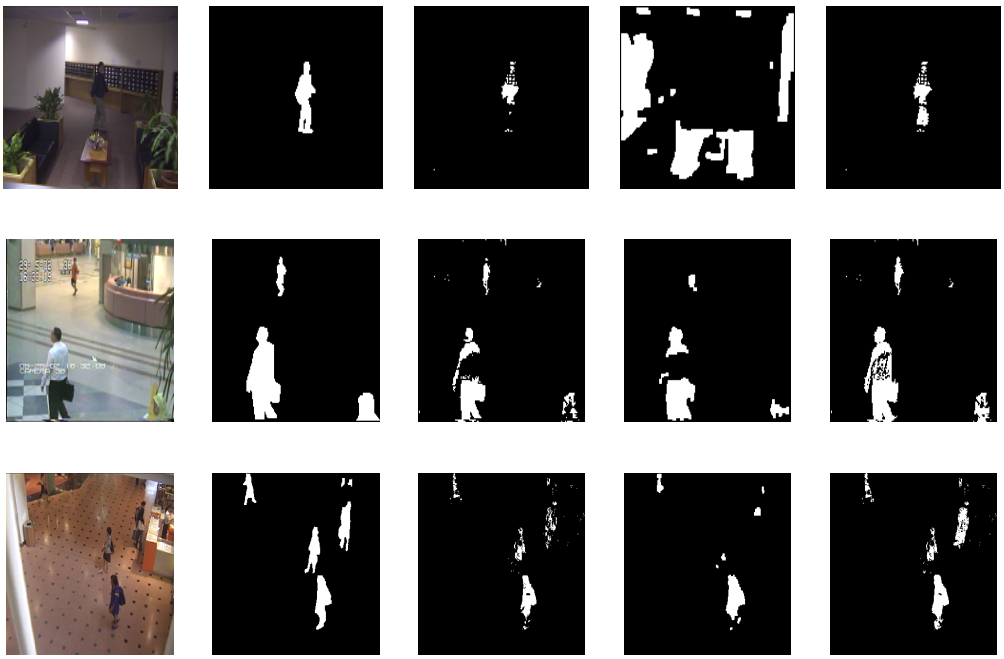
$$\text{and} label(P) = \begin{cases} FG & \text{if } (P = 1) \\ BG & \text{if } (P = 2) \end{cases} \tag{A.12}$$

The goal of this integration is to improve the detection rate of GMM background subtraction without deteriorating the precision. Actually, pixels belonging to the background and undergoing changes are correctly classified as background entities by GMM. However, these pixels are prone to be classified as foreground entities using optical flow. Therefore, we chose to start with the labels of GMM, then, by using the measure defined for uniform motion, the label of each pixel is updated. This integration adds spatial and temporal coherence since labeling process using GMM is done only on pixel level. It is an efficient way to improve the results and to avoid outliers caused by optical flow as well. Experimental results reported in the next section will demonstrate the effectiveness of our proposed approach.

A.4 Experimental Results

The proposed algorithm is compared with the improved adaptive GMM [138], and the foreground object detection method [72]. To evaluate these methods, we used the i2r dataset (`http://perception.i2r.a-star.edu.sg/bk_model/bk_index.html`) with available ground truths. The dataset is composed of nine video sequences captured in challenging environments. For each sequence, ground-truth foreground masks are provided for 20 randomly selected frames.

Using this dataset, both of qualitative and quantitative analysis of the results are presented with comparisons to the already cited methods. Figure A.2 shows results on three frames from different sequences. The results of the proposed method are qualitatively better than those obtained by the other methods.



(a) (b) (c) (d) (e)

Figure A.2: *Foreground segmentation results (a) Evaluation frames (b) Ground-truth foreground masks (c) Results of improved adaptive GMM [138] (d) Results of foreground object detection method [72] (e) Results of our proposed approach*

For quantitative evaluation, we use these metrics to compare the foreground mask to the ground truth:

- Detection Rate: It is called also true positive rate or recall defined by:

$$DR = \frac{TP}{TP + FN} \tag{A.13}$$

- False Acceptance Rate: it corresponds to $1 - p$ where p is called precision, it is defined by:

$$FAR = \frac{FP}{FP + TN} \tag{A.14}$$

where TP, TN, FP and FN denote respectively the total number of true positives, true negatives, false positives and false negatives.

Video sequences	Metrics	Improved adaptive GMM [138]	Foreground object detection method [72]	Our proposed approach
Restaurant	recall	55.09	48.48	63.41
	precision	99.61	97.38	99.03
Curtain	recall	38.92	41.66	70.62
	precision	99.95	99.89	99.23
Escalator	recall	71.65	40.02	74.36
	precision	98.75	98.31	98.24
Fountain	recall	44.42	40.40	54.87
	precision	99.28	99.41	99.23
Water Surface	recall	67.72	50.05	79.39
	precision	99.77	99.19	99.44
Trees	recall	73.99	63.49	88.14
	precision	97.45	99.63	97.19
Shopping center	recall	52.18	59.50	66.60
	precision	99.69	98.33	99.28
Lobby	recall	40.14	38.87	73.34
	precision	99.97	94.1	99.88
Hall	recall	39.10	47.37	63.18
	precision	99.71	99.08	99.27

Table A.1: Quantitative evaluation of our proposed approach compared to other methods

Quantitative results using these metrics are reported in Table A.1. These results show that the improved GMM [138] reaches 99% for the precision (we compute the average of different results), however, the detection rate is neither sufficient nor satisfactory for many applications (only 53%). That is why, our proposed method showed substantial improvement over GMM by increasing the detection rate (by 17%) and the precision remains roughly the same (around 99%). Also, by means of comparison to the method proposed in [72], our method gives better results. For the detection rate, it achieved a noteworthy improvement of about 23% compared to [72]. For the precision, the method proposed in [72] achieved 98%.

From these comparisons, we conclude that our proposed method outperforms the other methods. In addition, as it is shown is Figure A.2, our results are able to detect full object or in a shape that can be useful in many other applications. Since foreground segmentation is a key step in automatic video surveillance, the superior results that we obtained can deeply affect many applications such as people detection and tracking, person counting, and so on.

A.5 Conclusion

In this Chapter, we proposed a new approach for robust and online foreground segmentation using Gaussian Mixture Model and motion cue. The proposed approach succeeds to harness the advantages of both cues by improving the detection rate without deteriorating the precision. Our approach has been also tested on dataset of complex background scenes. The results demonstrate its ability to improve significantly the accuracy of the foreground segmentation compared to other existing approaches in the literature. The obtained superior results are significant since many applications can be carried out after performing reliable foreground segmentation.

APPENDIX B

Résumé en Français

B.1 Introduction

B.1.1 Contexte et motivation

Une foule est une multitude de personnes, attroupées les unes près des autres en un même lieu. Ce phénomène a été étudié dans différentes disciplines comme la sociologie, l'urbanisme et la physique. Aujourd'hui, c'est un sujet de recherche attractif dans le domaine de la vision par ordinateur. On accorde à l'étude de la foule de plus en plus d'importance, notamment en raison du nombre croissant des événements populaires qui drainent un nombre élevé de personnes comme les supermarchés, les fêtes religieuses, les événements sportifs et les manifestations.

Dans ce contexte, l'analyse des scènes denses s'avère une tâche primordiale permettant de contrôler et de gérer les foules. On accorde une importance particulière à l'estimation de la densité de la foule, notamment pour assurer la sécurité de personnes. Cette estimation peut être utile pour anticiper et empêcher les débordements potentiellement dangereux, surtout lorsque le nombre de personnes dépasse certaines limites. Plusieurs tragédies illustrent ce problème lié à l'affluence d'un nombre élevé d'individus, comme les heurts qui se produisent dans les stades ou les festivals. Pour éviter telles conséquences néfastes, on doit identifier ces situations dangereuses et prendre les décisions adéquates afin d'assurer une assistance et un plan d'urgence. Outre son utilité dans le domaine de la sécurité, l'estimation de la densité s'avère pertinente aussi pour des applications dans le domaine économique comme l'organisation des emplois du temps des personnels dans les centres commerciaux et les horaires des services de transport public.

Pour résoudre les problèmes inhérents à la gestion des foules, de nombreux travaux ont été réalisés ces dernières années. L'objectif consiste à estimer le nombre de personnes ou le niveau de densité de la foule dans une séquence vidéo. D'une manière générale, vu que les méthodes basées sur la détection de personnes ne sont pas efficacement opérationnelles sur des vidéos avec une foule très dense, des méthodes plus sophistiquées ont été adoptées. Elles sont fondées sur l'apprentissage de la relation liant un ensemble de caractéristiques dites de « bas niveau » et le nombre des personnes ou le niveau de la foule.

Dans cette thèse, notre recherche se focalise sur l'estimation de la densité de la foule et son utilisation dans d'autres applications en vidéo surveillance. En particulier, notre travail de recherche a pour objectifs :

- De résoudre les problèmes de comptage de personnes et d'estimation du niveau de la foule dans des scènes très denses. Ces problèmes sont d'un intérêt crucial dans les systèmes de surveillance.

- D'étendre l'estimation de la densité de la foule au niveau pixel à travers la construction des cartes de densité.

- De démontrer que dans des scènes très denses, lorsque les algorithmes de l'état de l'art pour la détection et le suivi ne fonctionnent pas correctement, l'estimation de la carte de densité peut améliorer considérablement les résultats.

- D'étudier l'utilité d'utiliser la densité de la foule en contexte de vie privée.

- De prouver la pertinence de l'utilisation de la densité locale ainsi que des informations sur les mouvements pour la détection de tout changement et la reconnaissance des évènements dans la foule.

B.1.2 Contributions

Notre recherche est axée sur des problèmes liés à l'analyse de la densité de la foule. En particulier, deux éléments essentiels ont été étudiés dans la littérature, à savoir le comptage de personnes et l'estimation du niveau de la foule. S'agissant du premier volet de notre problématique, nous proposons une nouvelle méthode, fondée sur l'utilisation d'une seule caractéristique normalisée dans l'étape de régression. En s'appuyant sur cette démarche méthodologique, nous avons adopté deux approches : la première approche est basée sur des mesures des points d'intérêt où une normalisation en perspective et une mesure de la foule sont combinées en une seule entité avec le nombre des points d'intérêt SIFT en mouvement (ce travail a été publié à EUSIPCO, 2012). Ensuite, la corrélation entre cette caractéristique et le nombre de personnes est obtenue par une régression gaussienne. D'après les tests expérimentaux réalisés, notre approche a abouti à des résultats plus précis par rapport aux autres méthodes existantes.

Quant à la deuxième approche proposée (publiée à WIFS, 2012), nous y adoptons une intégration de GMM pour la soustraction de l'arrière-plan avec un modèle de mouvement uniforme. Cette intégration a l'avantage d'améliorer la séparation entre l'avant-plan et l'arrière-plan (ce travail a été publié à ICIEV, 2012). Puis, pour le comptage étant basé sur des mesures de pixels d'avant-plan, nous proposons d'appliquer une normalisation en perspective. De plus, nous appliquons une mesure de la foule en synthétisant des caractéristiques locales FAST dans une densité globale. En adoptant cette démarche, nous avons également obtenu de meilleurs résultats en comparaison avec d'autres méthodes existantes. Par ailleurs, dans ce travail de recherche, nous démontrons également les avantages de l'intégration de GMM avec le mouvement.

Après avoir étudié le problème de comptage de personnes en testant certaines caractéristiques statistiques (le nombre des points d'intérêt de SIFT, l'aire de la partie mouvante et la densité des coins), nous abordons la classification du niveau de la foule. Pour ce faire, d'abord, nous travaillons sur des régions d'intérêt. Ensuite, dans le bloc d'extraction des caractéristiques, nous étudions la capacité descriptive du Local Binary Pattern (LBP) pour l'estimation de la densité par rapport à d'autres caractéristiques de texture. Aussi, nous explorons l'impact de la réduction de la dimension (DR) du LBP en blocs (Ce travail a été publié aux ateliers ICME, 2013).

Il est à noter que l'estimation de la densité de la foule est un problème multi-classe dont le but est d'attribuer différents niveaux de la foule aux régions locales de l'image. Sachant que Support Vector Machine (SVM) est conçu pour la classification binaire, plusieurs SVM binaires doivent être effectuées pour résoudre ce problème. Pour maintenir une faible complexité de calcul, nous proposons un algorithme qui implique moins de classificateurs SVM binaires. Cet algorithme est basé sur la réévaluation de chaque classification binaire en utilisant des scores de pertinence (ce travail a été publié à ICIP, 2013). Les résultats montrent que les techniques de la réduction de la dimension améliorent considérablement les performances de classification par rapport aux caractéristiques brutes. De plus, en comparaison avec d'autres caractéristiques de texture, LBP + DR a donné des résultats plus précis. De même, la comparaison de l'algorithme de Multiclass SVM proposé avec deux méthodes de référence montre indéniablement l'utilité de notre algorithme en termes de précision tout en gardant un coût de calcul plus faible.

Après avoir étudié les deux formes de la densité de la foule (à savoir le comptage de personnes et le niveau de la densité), nous proposons une nouvelle approche dans laquelle l'information locale au niveau du pixel remplace un niveau global de la foule ou un nombre de personnes par image. Bien que les formes de comptage de personnes et la classification du niveau de la foule soient couramment utilisées dans des applications liées à la sécurité, ces méthodes ont l'inconvénient de fournir une information globale sur l'ensemble de l'image. Pour remédier à cet inconvénient, nous recourons donc à l'information de la foule au niveau local par le calcul des cartes de densité en utilisant des caractéristiques locales en tant qu'observation d'une fonction probabiliste de densité. L'approche proposée inclut également une étape de suivi des caractéristiques qui permet d'exclure les points d'intérêt liés à l'arrière-plan. Nous avons testé notre approche sur des vidéos de différentes bases de données: les résultats obtenus démontrent l'efficacité de l'utilisation des suivis des points d'intérêt pour l'estimation de la densité. En outre, nous avons mené une étude comparative entre des différentes caractéristiques locales (ce travail a été publié à MMSP, 2013).

Cela étant, nous explorons un domaine de recherche prometteur qui consiste à utiliser des mesures de la densité de la foule pour compléter d'autres applications de la vidéo surveillance, telles que l'amélioration de la détection et le suivi de personnes dans des scènes encombrées, la détection et la reconnaissance des événements dans la foule et le renforcement de l'équilibre entre la surveillance et la protection de la vie privée.

D'abord, nous proposons d'utiliser les cartes de densité pour améliorer la détection et le suivi de personnes dans des scènes denses où la distinction des individus est quasiment impossible en raison des problèmes d'occultation. L'idée est fondée sur l'intégration des informations supplémentaires sur les foules dans le détecteur tout en appliquant un paramétrage dynamique qui utilise la mesure de la densité. Notre approche inclut également un apprentissage auto-adaptatif du rapport humain et de la hauteur perçue pour réduire les détections de faux-positives. Les avantages de l'intégration de la densité et des contraintes géométriques dans le processus de détection ont été testés expérimentalement donnant ainsi de meilleurs résultats (ce travail a été publié à AVSS, 2013).

Assurément, l'obtention des détections fiables peut avoir un impact sur d'autres applications. À titre d'exemple, nous étendons notre algorithme de détection au suivi de personnes en appliquant le filtre Probability Hypothesis Density (PHD). Comme prévu, les résultats de cette méthode montrent une amélioration par rapport à la méthode de référence, sachant que les suivis s'appuient directement sur les détections (cette extension est soumise au Signal Processing Journal: Image Communication).

Ensuite, nous proposons d'utiliser les cartes de densité pour ajuster le niveau de protection de la vie privée. Plus précisément, nous construisons des filtres adaptatifs dans lesquels le niveau de protection diminue progressivement en fonction de la densité de la foule. L'idée est basée sur l'observation: moins la foule est compacte, plus un seul individu est identifiable. En même temps, pour des raisons de sécurité, les agents doivent avoir une information visuelle claire dans les zones denses où des événements potentiellement dangereux risquent de se produire. Il est donc important de réduire le niveau de protection de la vie privée dans les scènes denses par rapport à des scènes comportant peu d'individus (ce travail a été publié à DSP, 2013). Pour démontrer l'efficacité de ces filtres contextualisés, nous proposons une évaluation objective du compromis entre la protection de la vie privée et l'intelligibilité. Les tests effectués montrent que nos filtres conservent de bonnes performances sur les tâches d'intelligibilité telles que le comptage de personnes et leur détection. En même temps, ces filtres sont capables de réduire considérablement les performances d'appariement qui peuvent potentiellement exposer des informations personnelles des personnes filmées et mettre ainsi leurs vies privées en danger (ce travail a été publié à ISM, 2013).

La dernière application consiste à utiliser une mesure de la densité pour des applications de haut niveau, telles que la détection de changement et la reconnaissance des évènements au sein de la foule. Bien que la plupart des méthodes existantes dans ce domaine s'appuient sur des informations de mouvement comme la vitesse et la direction, nous considérons que la densité locale est également importante. Notre approche est fondée sur l'extraction des répartitions locales de personnes ainsi que sur des informations de mouvement en utilisant les trajectoires des caractéristiques locales. Les résultats expérimentaux démontrent l'efficacité de notre approche pour une détection précoce des changements de comportement de la foule et des résultats précis pour la reconnaissance d'événements (ce travail est

soumis à ICPR 2014 et son extension est soumise au Signal, Image and Video Processing Journal, Special issue on Semantic representations for social behavior analysis in video surveillance systems).

B.1.3 Plan

Le travail présenté dans cette thèse s'inscrit dans l'analyse de la densité de la foule et de son utilisation dans d'autres applications liées à la vidéo surveillance. Nous pouvons répartir nos contributions dans le domaine de l'analyse de la densité dans deux grandes parties:

1. Dans la première phase de cette thèse, nous axons notre recherche sur les problèmes liés à l'estimation de la densité en abordant le comptage de personnes, l'estimation du niveau de la densité et la segmentation de la foule à l'aide des caractéristiques de bas niveau.

    - Dans la section B.2.1, nous proposons une nouvelle solution de comptage des personnes où seulement une caractéristique normalisée est utilisée dans l'étape de régression. Pour atteindre cet objectif, nous avons utilisé la distance et la densité de la foule. Le premier facteur est utilisé pour résoudre le problème de distorsions de perspective, alors que le deuxième facteur est utilisé pour détecter et mesurer le recouvrement entre individus.
    - Dans la Section B.2.2, nous étudions le problème de la classification du niveau de la foule. En particulier, notre étude est focalisée sur la capacité descriptive des caractéristiques LBP ainsi que sur l'impact de la réduction de dimension sur les caractéristiques en question. De plus, nous proposons une nouvelle solution Multiclass SVM qui maintient une faible complexité de calcul.
    - Dans la Section B.2.3 nous proposons un modèle spatio-temporel de la densité en utilisant les suivis des caractéristiques comme des observations d'une fonction probabiliste. Comparée aux autres méthodes communément utilisées (c.à.d. le nombre de personnes et le niveau de la foule), cette mesure a l'avantage de fournir une information locale sur la densité. C'est la raison pour laquelle elle sera utilisée dans d'autres applications que nous présentons dans la seconde partie de cette thèse.

2. Dans la deuxième phase de ce travail de recherche, nous montrons comment une estimation supplémentaire de la densité peut fournir des informations précieuses et compléter d'autres applications de vidéo surveillance. En particulier, trois applications sont explorées:

    - Dans la Section B.3.1, nous présentons notre approche pour améliorer la détection et le suivi de personnes dans des scènes denses. Cette approche est basée

sur l'intégration de la densité tout en ajoutant des contraintes géométriques dans le processus de détection et de suivi.

- Dans la Section B.3.2, nous proposons une nouvelle application de la densité par rapport à la question de la vie privée. Le concept de la protection de la vie privée qui tient compte du contexte a émergé récemment, vu que le niveau de protection nécessaire est profondément lié à l'activité de surveillance. L'efficacité des filtres contextualisés proposés a été démontrée par une évaluation objective du compromis entre l'intelligibilité et la protection de la vie privée.
- Dans la section B.3.3, nous proposons une nouvelle approche pour détecter les changements et reconnaître les événements dans la foule. Elle est fondée sur l'analyse des répartitions spatiales et temporelles de personnes en utilisant des trajectoires à long terme.

B.2 Analyse des caractéristiques de bas niveau pour l'estimation de la densité des foules

B.2.1 Comptage des personnes à l'aide d'une caractéristique normalisée

B.2.1.1 Caractéristique normalisée

Pour effectuer le comptage de personnes, nous appliquons la méthode de la régression des caractéristiques. L'avantage majeur de cette méthodes est qu'elle est indépendante des étapes intermédiaires de détection et de suivi des individus. Dans notre recherche, nous cernons les facteurs qui influent sur la relation entre les caractéristiques et le nombre de personnes. Plus précisément, nous explorons deux facteurs qui sont la distance par rapport à la caméra et la densité de la foule. Pour atteindre cet objectif, une normalisation en perspective et une mesure de la foule sont appliquées afin de compenser les variations de distance et de densité. Ces deux normalisations sont introduites dans une seule caractéristique normalisée. Cette démarche vise à rendre la caractéristique invariable vis-à-vis des facteurs mentionnés ci-dessus. Les deux normalisations sont détaillées comme suit :

- Normalisation en perspective: l'objectif de cette normalisation est de compenser les changements du nombre de caractéristiques extraites dûs à la distorsion de perspective. Les effets de perspective peuvent être simplement expliqués par le fait que les objets loin de la caméra apparaissent plus petits que ceux qui sont plus proches. Ce problème peut être résolu en pondérant chaque caractéristique selon un plan de perspective en affectant un plus grand poids pour les points les plus éloignés de la scène. En conséquence, différents poids W_p sont attribués en fonction de l'ordonnée y.
- Normalisation en densité: outre les distorsions de perspective, les caractéristiques

extraites sont aussi extrêmement sensibles au niveau de la densité. Quand les personnes sont plus proches les unes des autres, moins de points sont extraits à cause des occultations. Pour résoudre ce problème, nous proposons de synthétiser des caractéristiques locales dans une densité globale. Ensuite, notre recherche vise à formuler une fonction de pondération en utilisant la densité des caractéristiques locales comme une mesure de la foule. Plus précisément, notre objectif est de pondérer les caractéristiques en amplifiant leurs valeurs dans le cas d'une foule dense et en les réduisant dans le cas contraire. Pour ce faire, nous utilisons les valeurs de densité estimée d_k, $k = 1...M$, où M est le nombre total d'images dans les séquences vidéo utilisées pour l'apprentissage. Puis, nous définissons la fonction de poids d'un échantillon i en tant que :

$$W_d(i) = \frac{d_i - \mu}{\sigma_{max}} + 1 \tag{B.1}$$

ou $\mu = \frac{1}{M}\sum_{k=1}^{M} d_k$ et σ_{max} est le maximum d'écart-type σ_k.

Dans les paragraphes suivants, nous présentons nos approches pour le comptage de personnes basées sur des mesures de points d'intérêt et des mesures de pixels d'avant-plan.

Approche basée sur des mesures des points d'intérêt: Pour décrire le contenu de chaque image en cours d'analyse, on détecte seulement des points d'intérêt [76]. Ensuite, on affecte l'information de mouvement [39] à ces points détectés pour distinguer les points mobiles de ceux qui sont statiques. Après, pour prendre en compte les effets de distorsions de perspective, on applique les poids W_p (définis ci-dessus) en fonction de l'ordonnée y de chaque point d'intérêt. Afin d'estimer la densité des points d'intérêt, on utilise un algorithme de répartition. Cette démarche est importante pour pouvoir distinguer les points d'intérêt appartenant à des groupes de personnes différents. La solution la plus appropriée à ce problème est le « clustering » basé sur la densité où les groupes sont identifiés en fonction de la densité spatiale des points. Cette technique a également l'avantage d'être suffisamment flexible pour découvrir les clusters ayant une forme arbitraire. Dans l'étape suivante, les contours de chaque cluster sont définis en utilisant la technique α-shape. Puis, on calcule la densité en divisant le nombre des points d'intérêt en mouvement par l'aire totale des clusters. Ainsi en s'appuyant sur la densité estimée on calcule la fonction de poids définie en (B.1). La caractéristique proposée basée sur des mesures des points d'intérêt est définie par :

$$FeatN^1_{p,d}(i) = W_d(i) * \sum_{y=1}^{Y} W_p(y) * N_T(y) \tag{B.2}$$

où $N_T(y)$ est le nombre total des points d'intérêt à la ligne y.
Figure B.1 montre le schéma d'extraction de la caractéristique normalisée basée sur des mesures des points d'intérêt.

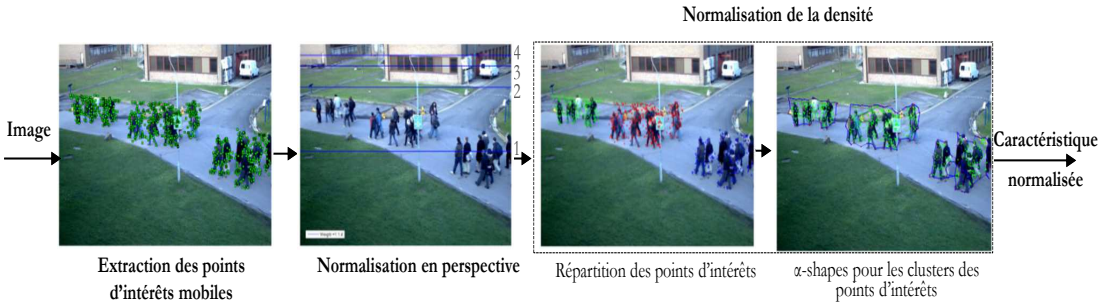


Figure B.1: *Schéma d'extraction de la caractéristique normalisée basée sur des mesures des points d'intérêt*

Approche basée sur des mesures des pixels en avant-plan: En tenant compte de l'importance de la segmentation de l'avant-plan et de son impact sur les prochaines étapes, nous avons adopté une solution efficace, basée sur l'intégration de GMM pour la soustraction de l'arrière-plan avec l'information de mouvement [46]. Ensuite, nous avons pris en compte seulement deux caractéristiques globales: le nombre des pixels de l'avant-plan et la densité des coins. La première caractéristique globale est pondérée selon le plan de perspective. Après, on estime la densité des coins en calculant le rapport entre le nombre des caractéristiques locales FAST et le nombre des pixels de l'avant-plan. En utilisant les valeurs de densité, on calcule la fonction de pondération définie en (B.1), ainsi la caractéristique normalisée est:

$$FeatN_p^2(i) = \sum_{y=1}^{Y} W_p(y) * FG_T(y) \tag{B.3}$$

ou $FG_T(y)$ est le nombre total des pixels en avant-plan à la ligne y.
Figure B.2 montre le schéma d'extraction de caractéristique normalisée basée sur des mesures des pixels en avant-plan.

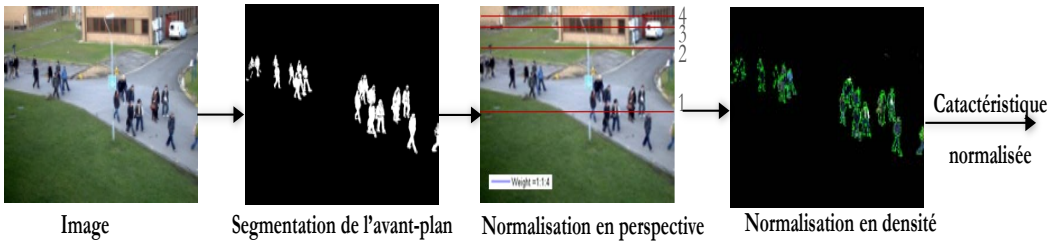


Figure B.2: *Schéma d'extraction d'une caractéristique normalisée basée sur des mesures des pixels en avant-plan*

B.2.1.2 Régression gaussienne

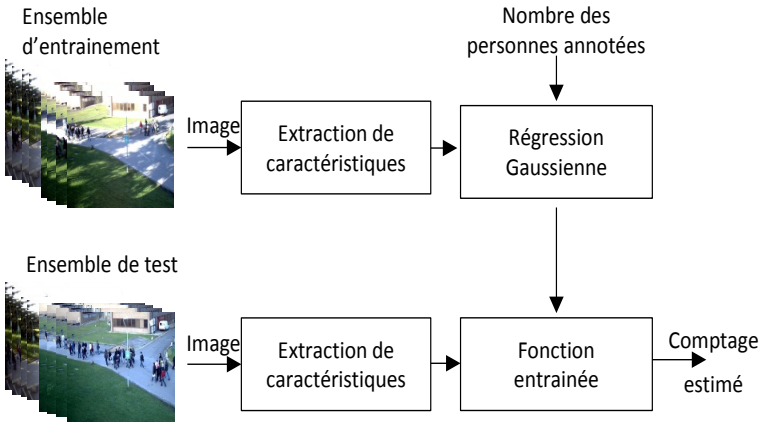


Figure B.3: *Organigramme du système de comptage*

Les deux caractéristiques proposées, basées sur des mesures des points d'intérêt et des pixels de l'avant-plan, définies respectivement dans (B.2), et (B.3) ont été formulées pour être invariantes par rapport à la perspective et à la densité de la foule. Ceci peut assurer la linéarité de la fonction liant les caractéristiques au nombre des personnes. Pour plus de flexibilité, il convient d'envisager d'éventuelles erreurs qui pourraient se produire dans la segmentation de la foule ou dans toute autre étape de notre système de comptage. Par conséquent, nous proposons d'utiliser une régression gaussienne qui est bien adaptée aux éléments linéaires avec des non-linéarités locales [96]. L'architecture de l'ensemble du système est illustrée dans la figure B.3.

B.2.1.3 Résultats expérimentaux:

Pour les résultats expérimentaux, nous utilisons la base de données PETS [41]. En particulier, nous utilisons vue 1 et vue 2 de la première section S1. Nous évaluons nos résultats selon les métriques MAE et MRE et nous comparons notre approche à deux autres méthodes existantes, voir figure B.4. En s'appuyant sur ces résultats, nous pouvons constater la différence de performance de la méthode de Albiol [3] entre la première et la seconde vue; cela pourrait justifier l'incapacité de cette méthode d'analyser des vidéos complexes, étant donné que la seconde vue comporte plus d'effets de perspective et une haute densité de personnes. Aussi, une comparaison de nos résultats avec la méthode de Conte [23] révèle les effets des normalisations proposées.

Pour l'évaluation de la deuxième approche, nous avons obtenu presque les mêmes résultats que pour la première: MAE est égale à 25 pour toutes les vidéos. Aussi, nous avons démontré l'impact de l'étape de segmentation sur le comptage en comparant les résultats entre l'application de GMM [138] et l'application de notre intégration de GMM avec le

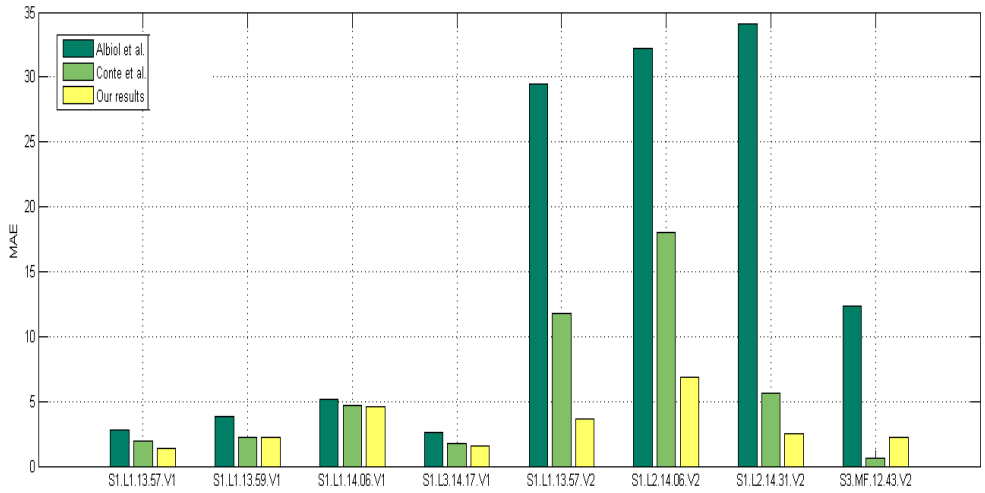


Figure B.4: *Évaluation quantitative de notre approche basée sur des mesures des points d'intérêt par rapport à d'autres méthodes*

mouvement [46], voir figure B.5. Cette comparaison montre une meilleure performance en utilisant l'intégration proposée. De même, nous démontrons l'efficacité de notre approche en prouvant que les deux normalisations permettent d'améliorer considérablement la précision des résultats de comptage, voir figure B.6.

B.2.2 Estimation du niveau de la foule par la classification des caractéristiques de texture

Afin de résoudre le problème de l'estimation du niveau de la foule, la classification proposée par Polus [94] est fréquemment adoptée. Selon cette méthode, la densité de la foule est classée en 5 niveaux: nulle, limitée, dense, très dense et surchargée (voir figure B.7). Dans ce travail de recherche, nous proposons d'effectuer l'estimation de la densité de la foule au niveau des régions, car en permettant à la fois la détection et la localisation des zones denses, cette méthode est plus appropriée que celle réalisée au niveau de l'image. Pour compenser les effets de distorsions de perspective au niveau des tailles des régions, on transforme les coordonnées dans le plan image en coordonnées réelles. Puis, on extrait des régions qui ont la même taille dans les coordonnées réelles. Ensuite, notre objectif consiste à affecter un niveau de la foule à chaque région. Pour atteindre ce but, il y a deux problèmes importants à résoudre, à savoir l'extraction des caractéristiques et la classification.

B.2.2.1 Réduction de dimension de LBP

Pour décrire le contenu de chaque région en cours d'analyse, les caractéristiques de texture sont extraites en utilisant la réduction de la dimension sur LBP en blocs. Ces dernières

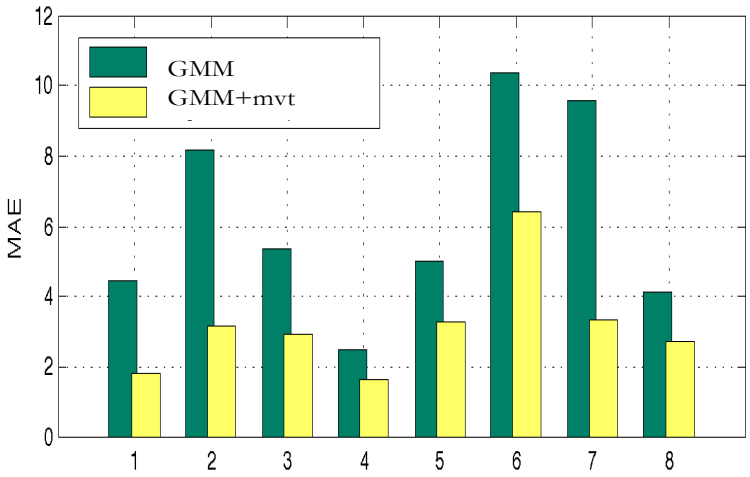


Figure B.5: *Comparaison des résultats de comptage en utilisant l'intégration de GMM avec le mouvement aux résultats basés seulement sur GMM*

années, LBP est fréquemment utilisé dans notre domaine car il s'agit d'un descripteur fiable permettant de caractériser la texture locale. Dans notre approche, d'abord, chaque région de l'image est divisée en plusieurs blocs à partir desquels les codes de LBP sont calculés. LBP en blocs est utilisé pour mieux préserver l'information locale. Ensuite, l'histogramme de chaque bloc est définie à travers l'extraction des occurrences des codes LBP. Enfin, les histogrammes calculés à partir des différents blocs sont concaténés. Pour M blocs, $\{B_1, B_1, .., B_M\}$, l'histogramme de chaque patch de l'image est formulé comme suit :

$$H = ((h_0^1, h_1^1, ..., h_{L-1}^1), ..., (h_0^M, h_1^M, ..., h_{L-1}^M)));$$
$$h_l^j = \sum_{(x,y)\in B_j} f\{LBP(x,y) = l\} \tag{B.4}$$

ou $[0, ..., L-1]$ dénote les niveaux du gris dans LBP. En tenant compte des différentes tailles de régions, il importe d'appliquer la normalisation suivante:

$$u = \sqrt{H/(||\ H\ ||_1 + \varepsilon)} \tag{B.5}$$

Un aperçu de l'extraction de LBP en blocs et de la normalisation de l'histogramme est illustré dans la figure B.8.

Le vecteur des caractéristiques de LBP extrait d'une région représente une grande dimension (de taille L x M) qui a un effet néfaste sur les étapes de la modélisation et de la classification. Par ailleurs, le vecteur des caractéristiques comporte des composants qui ne sont pas pertinents pour la densité de la foule et qui pourraient même avoir un effet négatif sur la performance de la classification. Ces sont les raisons pour lesquelles nous avons recours à des techniques de réduction de dimension. Plus précisément, on projette

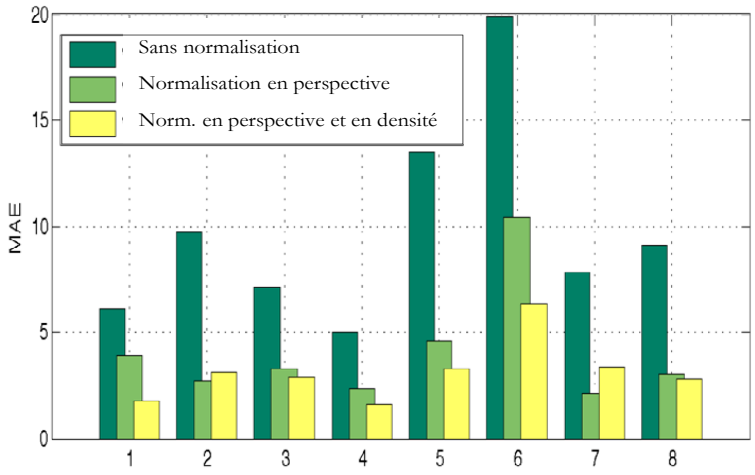


Figure B.6: *Amélioration apportée par la normalisation des pixels d'avant-plan par rapport aux distorsions en perspective et aux variations de densité de la foule*

les vecteurs des caractéristiques dans un espace discriminant en utilisant LDA sur le sous-espace PCA.

$$v = W_{LDA} W_{PCA} u \tag{B.6}$$

B.2.2.2 Multi-Classification SVM

Une fois les techniques de réduction de dimension sont appliquées sur LBP en blocs, les vecteurs des caractéristiques qui en résultent sont classés suivant différents niveaux de foule en utilisant Support Vector Machine (SVM) [24]. Sachant que l'estimation de la densité de la foule implique une classification multi-classes et que SVM est à l'origine un algorithme de classification binaire, le problème est résolu en combinant plusieurs classificateurs binaires. Les techniques les plus fréquemment utilisées sont: un-contre-un et un-contre-reste, où pour un problème à k-classes, k, et k (k - 1) / 2 classificateurs binaires sont respectivement effectués. À ce stade, notre recherche vise à améliorer la précision de la classification tout en gardant un coût faible de calcul par rapport aux approches existantes. L'algorithme que nous proposons est composé de (k - 1) classificateurs binaires. L'idée principale est de réévaluer chaque classificateur binaire en utilisant des scores de pertinence. En d'autres termes, nous dépassons la subdivision binaire en attribuant différents niveaux de la foule aux échantillons déjà classés. Cette division automatique est réalisée via un score flou qui est défini comme la probabilité à posteriori: $\sigma_s = p(l_s = sign(f(v_s))|f(v_s))$ avec un modèle paramétrique basé sur une fonction sigmoïde:

$$\sigma_s = \frac{1}{1 + \exp(af(v_s) + b)} \tag{B.7}$$

(a) nulle (b) limitée (c) dense

(d) très dense (e) surchargée

Figure B.7: *Définition des différents niveaux de la foule selon la densité*

Selon ces scores de pertinence, lors de l'apprentissage, les échantillons positifs et négatifs d'apprentissage de chaque classificateur sont triés. Puis, nous définissons différents seuils de telle sorte que pour les échantillons de chaque classe différents niveaux de la foule peuvent être affectés.

B.2.2.3 Résultats expérimentaux:

L'approche proposée est évaluée avec la base de données PETS [41]. D'abord, nous sélectionnons quelques régions des Sections S1 et S2. Ensuite, nous définissons les différents niveaux de la foule selon le nombre de personnes dans $13m^2$.

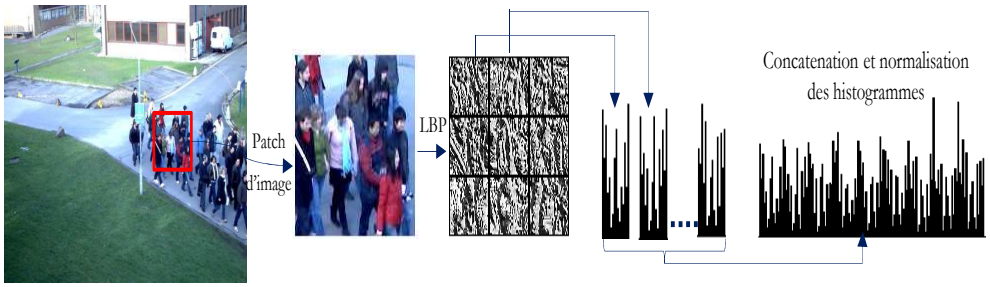


Figure B.8: *Extraction de LBP en blocs et la normalisation de la séquence histogramme*

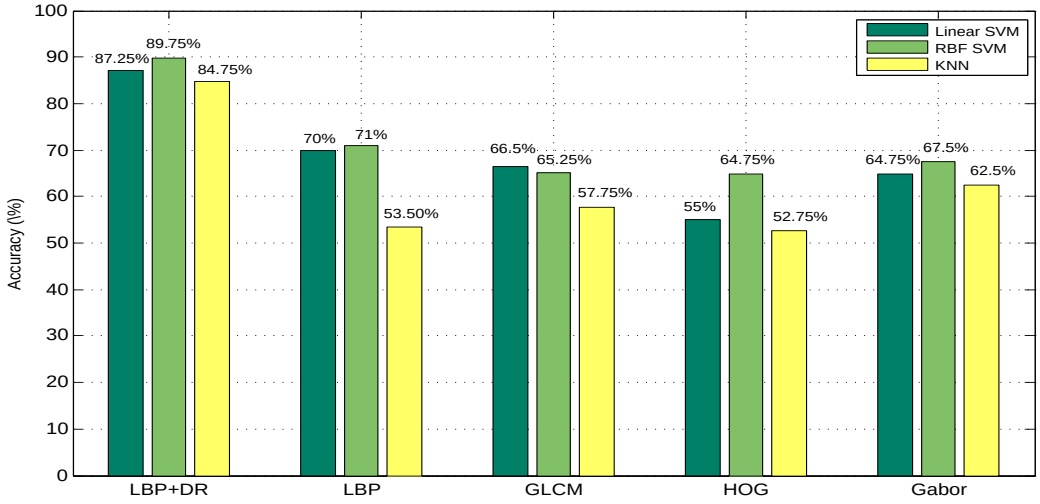


Figure B.9: *Comparaisons LBP+DR avec d'autres caractéristiques de texture (LBP, GLCM, HOG et Gabor) en utilisant un-vs-un SVM (pour les noyaux linéaires et RBF) et le classificateur KNN*

Dans la figure B.9, nous comparons les caractéristiques proposées (LBP+DR) avec d'autres caractéristiques de texture (LBP, HOG, Gabor et GLCM). Pour la classification, nous mettons en contraste SVM (pour les noyaux RBF et linaire) et KNN. Comme montré dans cette figure, la comparaison met en évidence une amélioration remarquable (20% pour RBF), due à la réduction de la dimension de LBP. En outre, cette comparaison montre que (LBP + DR) a une meilleure performance par rapport aux autres caractéristiques de texture. Quant aux classificateurs, SVM se montre plus performant par rapport à KNN (avec de meilleures performances pour le noyau RBF).

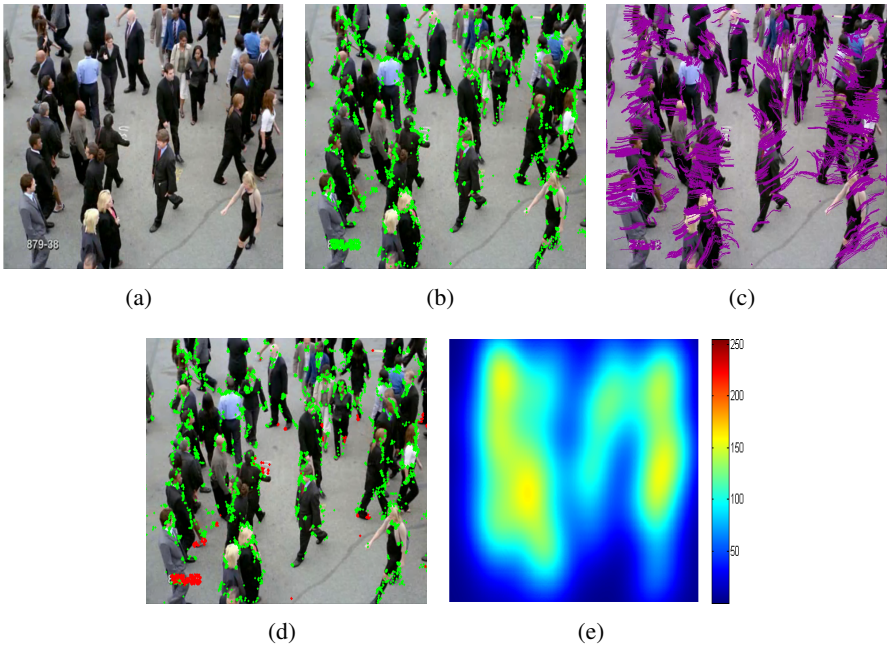
Dans le tableau B.1, la précision de la classification en utilisant notre algorithme de multiclasses SVM est comparé à un-contre-un et un-contre-reste. Nous ajoutons également une comparaison entre ces méthodes en termes de nombre des classificateurs binaires. D'après les résultats, l'algorithme proposé nécessite moins de classificateurs binaires. De plus, son évaluation en termes de précision montre de meilleurs résultats par rapport à un-contre-reste et des résultats comparables à un-contre-un.

Méthodes de multi-classes SVM	SVM (linéaire)	SVM (RBF)	Nb. de classificateurs binaires
Un-contre-un	87.25%	89.75%	6
Un-contre-reste	72.25%	84.00%	4
Algorithme proposé	88.25%	89.00%	3

Table B.1: Comparaison de l'algorithme proposé pour multiclasses SVM avec un-contre-un et un-contre-reste

B.2.3 Estimation de la carte de densité en utilisant le suivi des caractéristiques locales

Comme expliqué précédemment, la génération de la densité locale de la foule est plus utile qu'une densité globale ou qu'un certain nombre de personnes dans une image. Une illustration des modules de la carte de densité proposée est représentée dans la figure B.10. Dans ce qui suit, nous présentons notre approche pour l'estimation de la densité:



(a) (b) (c)

(d) (e)

Figure B.10: Illustration de la carte de densité proposée en utilisant le suivi des caractéristiques locales: (a) l'image testée (b) les caractéristiques locales FAST (c) le suivi des caractéristiques (d) distinction entre caractéristiques en mouvement (vert) et statiques (rouge) (e) estimation de la carte de densité de la foule

B.2.3.1 Extraction des caractéristiques locales

En s'appuyant sur l'hypothèse que les régions de faible densité présentent des caractéristiques locales moins denses par rapport aux régions de haute densité, nous proposons d'utiliser des caractéristiques locales comme une description de la foule en reliant leur densité à l'intensité de la foule. Concernant les caractéristiques locales, nous choisissons Features from Accelerated Segment Test (FAST) [104] pour les raisons suivantes: FAST a été proposé à l'origine pour la détection des coins. Ce détecteur a l'avantage d'être capable de trouver des petites régions qui sont sensiblement différentes de leurs pixels voisins. En outre, FAST a été utilisé dans [13] pour détecter les foules à partir d'images aériennes et les résultats obtenus montrent une détection fiable des régions denses. On compare les

performances de ces caractéristiques à Scale-Invariant Feature Transform (SIFT) [76], et à Good Features to Track (GFT) [117].

B.2.3.2 Le suivi des caractéristiques locales

L'utilisation des caractéristiques extraites pour estimer la carte de densité sans aucun processus de sélection pourrait provoquer deux problèmes: d'abord, le nombre élevé des caractéristiques locales augmente le temps de calcul de la densité. Aussi, les caractéristiques locales comportent des composants non pertinents pour l'estimation de la densité. Ainsi, nous avons besoin d'ajouter dans notre système une étape de séparation entre les caractéristiques statiques et en mouvement. Cela se fait en affectant des informations de mouvement pour les caractéristiques détectées en utilisant Robust Local Optical Flow [111]. En outre, nous établissons un système de vérification avant-arrière où la position résultante d'un point est utilisée comme entrée à la même étape d'estimation de mouvement à partir de la deuxième image. Après cette projection, les points pour lesquels le mouvement inverse ne retrouve pas les positions initiales sont rejetés. Pour les points restants, les informations de mouvement sont étendues pour former des trajectoires en reliant les vecteurs de mouvement calculés sur des images consécutives:

$$\begin{aligned} \mathcal{T}_k &= \{T_1^k, ..., T_{p_k}^k | \\ T_i^k &= \{X_i(k-\Delta t_i^k), Y_i(k-\Delta t_i^k), ..., X_i(k), Y_i(k)\}\} \end{aligned} \tag{B.8}$$

ou Δt_i^k est l'intervalle temporel entre la première et l'image actuelle d'une trajectoire T_i^k. $(X_i(k-\Delta t_i^k), Y_i(k-\Delta t_i^k)$, et $(X_i(k), Y_i(k))$ sont les coordonnées d'une caractéristique depuis son apparition et sa position actuelle. L'avantage de l'utilisation des trajectoires au lieu de calculer les vecteurs de mouvement entre deux images consécutives c'est que les valeurs aberrantes sont filtrées et l'ensemble des informations de mouvement est moins affecté par le bruit.

B.2.3.3 Estimation de la densité par noyau

Les trajectoires générées sont ainsi utilisées pour éliminer les caractéristiques statiques en comparant la moyenne de mouvement Γ_i^k de chaque trajectoire T_i^k à un certain seuil ζ qui est fixé selon la résolution de l'image et la perspective de la caméra. Ensuite, les caractéristiques en mouvement sont identifiées par la relation $\Gamma_i^k > \zeta$, alors que les autres sont considérées comme faisant partie de l'arrière-plan. L'utilisation de trajectoires est avantageuse parce que la séparation entre arrière-plan / avant-plan est améliorée et les positions des caractéristiques sont soumises à une étape de filtrage temporel.

Après avoir filtré les caractéristiques statiques, la carte de densité est définie par la fonction de densité de probabilité (pdf) qui est estimée en utilisant un noyau gaussien. Pour m_k caractéristiques extraites aux positions $\{(x_i, y_i), 1 \leq i \leq m_k\}$, la carte de densité

est définie comme suit:

$$C_k(x,y) = \frac{1}{\sqrt{2\pi}\sigma}\sum_{i=1}^{m_k} \exp -(\frac{(x-x_i)^2+(y-y_i)^2}{2\sigma^2}) \tag{B.9}$$

où σ est la bande passante du noyau gaussien.

B.2.3.4 Résultats expérimentaux

La carte de densité proposée est évaluée dans des scènes denses. En particulier, nous sélectionnons quelques vidéos de PETS [41], UCF dataset [4], et Data Driven Crowd Analysis dataset [103]. La stratégie de l'évaluation est la suivante: nous considérons qu'une estimation précise des cartes de densité peut représenter adéquatement les distributions spatiales des individus dans la scène. Pour ce faire, nous définissons la densité de vérité de terrain comme une densité du noyau basée sur des détections annotées. Étant donné un ensemble d'annotations $\phi_k = \{\varphi_1^k, ..., \varphi_{l_k}^k\}$, $\varphi_i^k = \{xc_i^k, yc_i^k, h_i^k, w_i^k\}$, la densité de la vérité de terrain correspondante est définie comme:

$$G_k(x,y) = \sum_{i=1}^{l_k} \frac{1}{\sqrt{2\pi}\sigma_i^k} \exp -(\frac{(x-xc_i^k)^2+(y-yc_i^k)^2}{2{\sigma_i^k}^2}) \tag{B.10}$$

σ_i^k correspond à la taille de φ_i^k.

Ensuite, nous supposons qu'une représentation optimale de la densité peut être réalisée par simple pondération linéaire de la densité de vérité de terrain. Pour les cartes de densité estimées $\{C_1, ..., C_N\}$ et leurs densités de vérité de terrain correspondantes $\{G_1, ..., G_N\}$, nous estimons la transformation linéaire de C_i en G_i, $1 \leq i \leq N$, avec le minimum de disparités entre eux. Le paramètre Ω de cette transformation est:

$$\begin{gathered}\Omega = \underset{\omega}{\operatorname{argmin}}(\omega^T\omega + \lambda \textstyle\sum_{i=1}^{N} Dist(G_i(.), C_i'(.|\omega)), \\ C_i'(.|w) = w^T C_i(.)\end{gathered} \tag{B.11}$$

L'évaluation est réalisée en utilisant MAE (erreur moyenne absolue) entre G_k et les densités après la transformation linéaire C_k'. En outre, nous avons divisé les régions d'image en dense ($\mathcal{C}$) / non-dense ($\overline{\mathcal{C}}$), voir les résultats dans le tableau B.2.

Ces comparaisons montrent que l'étape de suivi des caractéristiques débouche sur une amélioration substantielle par rapport à la segmentation de l'arrière plan. Cela met en évidence l'avantage d'utiliser des trajectoires, car notre estimation est plus résistante au bruit et le mouvement d'ensemble est plus précis. En outre, en comparant les différentes caractéristiques locales, les résultats montrent que le choix des caractéristiques a un impact limité sur la performance si on considère toutes les régions de l'image. Toutefois, une marge plus importante entre FAST et les deux autres caractéristiques est visible dans les

Séquence vidéo	E	$E_{\overline{\mathcal{C}}}$	$E_{\mathcal{C}}$
S1.L1.13-57 (FAST):	**0.07** / 0.20	**0.05** / 0.18	**0.30** / 0.44
S1.L1.13-57(SIFT):	**0.07** / 0.15	**0.05** / 0.13	0.32 / 0.38
S1.L1.13-57 (GFT):	0.08 / 0.17	0.06 / 0.14	0.34 / 0.40
S1.L1.13-59 (FAST):	**0.04** / 0.12	**0.04** / 0.11	**0.13** / 0.30
S1.L1.13-59 (SIFT):	**0.04** / 0.09	**0.04** / 0.09	0.18 / 0.27
S1.L1.13-59 (GFT):	**0.04** / 0.11	**0.04** / 0.10	0.18 / 0.31
S1.L2.14-31 (FAST):	**0.09** / 0.24	**0.07** / 0.21	**0.21** / 0.41
S1.L2.14-31 (SIFT):	**0.09** / 0.20	**0.07** / 0.17	0.24 / 0.41
S1.L2.14-31 (GFT):	0.10 / 0.22	0.08 / 0.18	0.27 / 0.41
S2.L3.14-41 (FAST):	0.04 / 0.23	0.03 / 0.20	0.23 / 0.54
S2.L3.14-41 (SIFT):	**0.03** / 0.17	**0.02** / 0.13	**0.21** / 0.60
S2.L3.14-41 (GFT):	**0.03** / 0.18	**0.02** / 0.15	**0.21** / 0.58
UCF-879 (FAST):	**0.10** / 0.28	**0.10** / 0.28	**0.09** / 0.23
UCF-879 (SIFT):	0.26 / 0.37	0.25 / 0.36	0.33 / 0.38
UCF-879 (GFT):	0.14 / 0.31	0.14 / 0.31	0.17 / 0.33
INRIA-879-42(FAST):	**0.11** / 0.36	**0.09** / 0.38	**0.21** / 0.30
INRIA-879-42 (SIFT):	0.16 / 0.33	0.13 / 0.34	0.28 / 0.31
INRIA-879-42 (GFT):	0.13 / 0.34	0.10 / 0.36	0.24 / 0.31

Table B.2: Résultats de l'estimation de la densité de la foule en termes de MAE (E, $E_{\mathcal{C}}$ et $E_{\overline{\mathcal{C}}}$). Val1/Val2 sont les résultats de notre approche en utilisant le suivi des caractéristiques et GMM pour la soustraction arrière-plan.

régions denses (en utilisant $E_{\mathcal{C}}$). Ceci prouve que FAST a de bonnes performances pour la mesure de la foule.

B.3 Applications utilisant de la densité de la foule

B.3.1 Détection et suivi des personnes dans des scènes denses

B.3.1.1 Contraintes de la densité

Compte tenu des difficultés rencontrées lors de la détection des personnes dans des scènes denses, il importe d'inclure des informations supplémentaires sur les foules afin d'adapter le détecteur à ce type de situations. Dans cette section, il s'agit principalement de présenter une extension d'un détecteur de l'état de l'art [40] dans ces scènes. Nous proposons d'utiliser la carte de densité présentée dans la Section B.2.3 comme une information contextuelle pour optimiser le comportement du détecteur en sélectionnant un seuil de détection dynamique. Cette démarche est particulièrement importante, car des seuils faibles sont appropriés dans des scènes denses et des seuils plus élevés assurent moins de faux positifs dans les scènes moins denses. Il est donc souhaitable de trouver un moyen de régler automatiquement le seuil de détection en fonction de la probabilité que des personnes sont

présentes ou non dans une certaine position de l'image. En utilisant un intervalle pré-défini des seuils de détection donné par τ_{max}/τ_{min}, nous appliquons une règle linéaire adaptative de la densité pour sélectionner automatiquement le seuil:

$$\tau_{dyn} = \tau_{min} + (\tau_{max} - \tau_{min}) \cdot \hat{C}_k(d_j^k), j \in \{1...n_k\} \tag{B.12}$$

avec $\hat{C}_k(d_j^k)$ est la valeur moyenne des densités dans d_j^k.

$$\hat{C}_k(d_j^k) = \frac{\sum_{p=0}^{h_j^k-1} \sum_{q=0}^{w_j^k-1} C_k(x_j^k + p, y_j^k + q)}{w_j^k \cdot h_j^k} \tag{B.13}$$

B.3.1.2 Contraintes géométriques

Outre les contraintes de densité, nous proposons d'appliquer des contraintes géométriques dans une étape de filtrage pour supprimer les détections de fausse taille. Ceci est important, parce que si le score d'une telle détection est plus élevé que les scores des détections d'objets individuels, l'étape de suppression non-maximale (NMS) le gardera à la place des détections correctes. Pour atteindre cet objectif, nous proposons d'appliquer des pré-filtres en utilisant la hauteur perçue et le rapport d'aspect.

Étant donné un ensemble de régions d'intérêts $\mathcal{D}_k(\tau) = \{d_1^k, ..., d_{n_k}^k\}$, d'après [53], la relation entre la position d'une personne et sa taille perçue est:

$$h_j^k = \alpha_{k-1} \cdot y_j^k + \beta_{k-1}, j \in \{1...n_k\} \tag{B.14}$$

où les paramètres α_{k-1} et β_{k-1} sont calculés par régression. Aussi, le rapport d'aspect est défini par:

$$\gamma_{k-1} = median \left\{ \frac{w_j^i}{h_j^i} \right\}_{1 \le i \le (k-1), 1 \le j \le n_i} \tag{B.15}$$

Les paramètres α_{k-1}, β_{k-1}, et γ_{k-1} sont calculés à partir des détections acceptées $\{\mathcal{D}_1, ..., \mathcal{D}_{k-1}\}$ et sont mis à jour à chaque image.

B.3.1.3 Résultats expérimentaux

Pour les évaluations quantitatives des résultats de détection, nous utilisons CLEAR métriques [121]: Multi-Object Detection Accuracy (MODA) et Multi-Object Detection Precision (MODP), voir tableau B.3. Comme il est indiqué dans ce tableau, nous comparons la méthode de référence [40] en utilisant deux seuils de détection ($\tau_{min} = (-0.5)$ et $\tau_{max} = (-1.2)$) avec notre méthode en utilisant un seuil dynamique τ_{dyn}. Des tests supplémentaires sont effectués pour évaluer l'impact des filtres de correction. Comme le montrent les résultats finaux (dans la dernière colonne), notre méthode basée sur l'utilisation d'un seuil

Séquence vidéo	τ_{min}	τ_{max}	τ_{dyn}	Filtrage	τ_{dyn} + Filtrage
S1.L1.13-57 (FAST):			0.59 / 0.59		**0.63 / 0.63**
S1.L1.13-57 (SIFT):	0.48 / 0.65$^{(*)}$	0.36 / 0.57$^{(*)}$	0.59 / 0.60	0.48 / 0.66	0.61 / 0.63
S1.L1.13-57 (GFT):			**0.60 / 0.60**		0.62 / 0.63
S1.L1.13-59 (FAST):			**0.60 / 0.67**		0.60 / 0.68
S1.L1.13-59 (SIFT):	0.56 / 0.68$^{(*)}$	0.25 / 0.61$^{(*)}$	**0.60 / 0.67**	0.56 / 0.69	0.60 / 0.68
S1.L1.13-59 (GFT):			0.59 / 0.67		**0.61 / 0.68**
S1.L2.14-31 (FAST):			**0.40 / 0.59**		**0.47 / 0.63**
S1.L2.14-31 (SIFT):	0.33 / 0.63$^{(*)}$	0.09 / 0.57$^{(*)}$	**0.40 / 0.59**	0.32 / 0.65	**0.47 / 0.63**
S1.L2.14-31 (GFT):			**0.40 / 0.59**		**0.47 / 0.63**
S2.L3.14-41 (FAST):			**0.34 / 0.56**		**0.35 / 0.57**
S2.L3.14-41 (SIFT):	0.29 / 0.54$^{(*)}$	0.04 / 0.56$^{(*)}$	0.34 / 0.54	0.29 / 0.54	0.35 / 0.55
S2.L3.14-41 (GFT):			0.34 / 0.54		0.36 / 0.55
UCF-879 (FAST):			0.41 / 0.55		**0.59 / 0.58**
UCF-879 (SIFT):	0.44 / 0.58$^{(*)}$	0.34 / 0.54$^{(*)}$	0.42 / 0.55	0.41 / 0.62	0.57 / 0.58
UCF-879 (GFT):			**0.43 / 0.55**		0.58 / 0.58
INRIA879-42 (FAST):			**0.35 / 0.55**		**0.42 / 0.47**
INRIA879-42 (SIFT):	0.27 / 0.54$^{(*)}$	0.06 / 0.55$^{(*)}$	**0.35 / 0.55**	0.20 / 0.42	0.38 / 0.45
INRIA879-42 (GFT):			**0.35 / 0.55**		0.41 / 0.44

Table B.3: Résultats de détection en termes de MODA / MODP

de détection dynamique avec filtrage, donne de meilleurs résultats pour toutes les vidéos de test. Encore une fois, le choix des caractéristiques ne semble pas avoir un impact visible sur la performance, à part une légère amélioration en utilisant FAST par rapport aux autres caractéristiques.

Séquence vidéo	($\tau = 0.5$)	méthode proposée
S1.L1.13-57 (FAST):		63.64
S1.L1.13-57 (SIFT):	65.26$^{(*)}$	62.69
S1.L1.13-57 (GFT):		**61.06**
S1.L1.13-59 (FAST):		**62.36**
S1.L1.13-59 (SIFT):	64.81$^{(*)}$	64.61
S1.L1.13-59 (GFT):		64.05
S1.L2.14-31 (FAST):		**66.39**
S1.L2.14-31 (SIFT):	75.27$^{(*)}$	70.82
S1.L2.14-31 (GFT):		71.00
S2.L3.14-41 (FAST):		87.65
S2.L3.14-41 (SIFT):	88.19$^{(*)}$	88.44
S2.L3.14-41 (GFT):		**87.36**
UCF-879 (FAST):		86.89
UCF-879 (SIFT):	89.92	86.95
UCF-879 (GFT):		**86.46**
INRIA-879-42 (FAST):		**73.22**
INRIA-879-42 (SIFT):	81.15$^{(*)}$	75.55
INRIA-879-42 (GFT):		73.56

Table B.4: Résultats de suivi en terme d'OSPA-T.

Pour démontrer l'impact de l'amélioration des résultats de détection sur le suivi, nous utilisons le filtre PHD [126] dans un schéma de suivi par détection. Les résultats selon la distance OSPA-T [99] sont présentés dans le tableau B.4. Dans tous les cas, en comparaison avec la méthode de référence, les résultats obtenus en utilisant un seuil de détection

dynamique avec filtrage sont meilleurs. Ces résultats correspondent bien à nos attentes, puisque le suivi est fondé directement sur les résultats de détection.

B.3.2 L'analyse du comportement de la foule

Pour atteindre une meilleure performance dans l'analyse du comportement de la foule, nous considérons que les mesures de densité peuvent être une source d'information sur la répartition spatiale des personnes dans la scène. Ces informations sont utiles, notamment pour localiser et reconnaître les événements au sein de la foule telle que l'évacuation, la formation et la division de la foule. Par conséquent, dans notre approche, nous considérons simultanément ces deux mesures de la foule, à savoir l'apparence (densité) et le mouvement (vitesse et direction).

Les trajectoires définies dans (B.8) et qui sont utilisées dans une première étape pour estimer les cartes de densité, sont également employées dans une deuxième étape pour extraire des informations sur le mouvement de la foule. Dans cette étape, nous ne prenons en compte que les trajectoires à long terme, tandis que d'autres trajectoires à court terme sont filtrées. Aussi, nous limitons l'historique de chaque trajectoire 2D sur quelques images. Car dans le cas contraire, en considérant l'ensemble de la trajectoire, une augmentation de la vitesse ne sera pas détectée tôt. Aussi, la direction de mouvement peut être moins précise. Pour modéliser la foule, on code chaque attribut par 1D-histogramme: Nous quantifions la densité locale C_K en $N_d = 5$ bins, l'orientation des vecteurs de mouvement θ en $N_\theta = 8$ bins. Quant à la vitesse elle est quantifiée en $N_s = 5$ classes: très lent, marche, marche rapide, course et course rapide (nous corrigeons les effets de perspective pendant le calcul de la vitesse).

Après la modélisation des attributs par des histogrammes, leur application pour l'analyse du comportement de la foule est démontrée en deux étapes: d'abord, la variation dans le temps d'une mesure de la stabilité (en utilisant les histogrammes) est utilisée pour détecter les changements ou les événements anormaux, voir le paragraphe B.3.2.1. Ensuite, un vecteur des caractéristiques concaténant ces histogrammes est utilisé pour la reconnaissance des événements, voir le paragraphe B.3.2.2.

B.3.2.1 Détection de changement dans la foule

Selon la méthode décrite plus haut, nous avons $H_d(k)$, $H_\Theta(k)$ et $H_s(k)$ qui désignent, respectivement, les histogrammes de densité, l'orientation et la vitesse. Si un changement intervient dans le comportement de la foule, cela générerait des changements entre les histogrammes. Ainsi, nous comparons les histogrammes au cours du temps suivant la même stratégie que dans [29]: nous calculons la stabilité temporelle $\sigma_i(k)$ de chaque histogramme

$h_i(k)$ comme la moyenne pondérée d'un vecteur de similarité $S_i(k)$:

$$\sigma_i(k) = \omega^T S_i(k),$$
$$\omega = \frac{1}{\sum_{j=1}^{n} e^{\lambda \Delta t_j}} (e^{-\lambda \Delta t_1}, e^{-\lambda \Delta t_2}, ..., e^{-\lambda \Delta t_n}) \tag{B.16}$$

λ représente la constante de décroissance, $\Delta t_j = j\Delta t$ (Δt est une constante). $S_i(k)$ est calculé en utilisant la métrique de corrélation entre chaque histogramme $H_i(k)$ les histogrammes de n images précédentes $H_i(k - \Delta t_1)$, ..., et $H_i(k - \Delta t_n)$.

Selon notre approche, un changement est détecté si la stabilité temporelle pour un attribut est faible. Pour ce faire, nous comparons chaque stabilité temporelle $\sigma_i(k), 1 \leq i \leq 3$ avec un seuil adaptatif $\tau_i(k)$ calculé comme la moitié de la moyenne des σ_i entre $(k - \Delta t_1)$ et $(k - \Delta t_n)$:

$$\tau_i(k) = \frac{1}{2n} \sum_{j=1}^{n} \sigma_i(k - \Delta t_j) \tag{B.17}$$

B.3.2.2 Reconnaissance des événements dans la foule

Pour la reconnaissance, 6 événements sont testés, à savoir la marche, la course, l'évacuation, la dispersion locale, la formation et la division de la foule. Étant donné une image **x**, nous visons à la classer dans l'un des événements $v^* \in \mathcal{V}$, qui maximise la probabilité conditionnelle:

$$v^* = \arg\max_{v \in \mathcal{V}} P(v|\mathbf{x}, \theta^*) \tag{B.18}$$

où θ^* sont estimés lors de l'apprentissage. Ceci peut être réalisé par la classification SVM. S'agissant du vecteur des caractéristiques, nous concaténons les 3 histogrammes $H_d(k)$, $H_\Theta(k)$, et $H_s(k)$ dans $\mathcal{H}_k$. Pour la classification, on utilise le noyau Chi-Square:

$$K(\mathcal{H}_i, \mathcal{H}_j) = \sum_I \frac{\mathcal{H}_i(I) - \mathcal{H}_j(I))^2}{\mathcal{H}_i(I) + \mathcal{H}_j(I)} \tag{B.19}$$

B.3.2.3 Résultats expérimentaux

Pour détecter le changement au sein de la foule, nous testons notre approche sur la base UMN [1], qui a été largement utilisée pour distinguer les activités normales et celle qui sont anormales au sein de la foule. Pour l'évaluation quantitative, nous employons l'erreur relative moyenne [64], voir tableau B.5.

Comme le montre le tableau B.5, la comparaison de nos résultats de détection avec la vérité terrain montre des détections plus précises dans la plupart des vidéos. Le retard dans la détection de certaines images après l'événement s'explique par notre stratégie de détection selon laquelle un événement anormal est détecté uniquement si la stabilité dans le temps est inférieure au seuil dynamique. Un changement nécessite donc certain temps pour

seq. UMN	nb. de frames	vérité terrain	nos résultats	e_F
Video1	625	484	493	0.0144
Video2	828	665	669	0.0048
Video3	549	303	319	0.0291
Video4	685	563	582	0.0277
Video5	769	492	512	0.0260
Video6	579	450	466	0.0276
Video7	895	734	754	0.0223
Video8	667	454	471	0.0255
Video9	658	551	551	0
Video10	677	570	577	0.0103
Video11	807	717	722	0.0062

Table B.5: Comparaison de nos résultats de détection avec une vérité de terrain en utilisant l'erreur relative moyenne

être détecté, ce qui justifie ce retard. Par contre, cette stratégie est adéquate pour éviter les fausses alertes.

Pour évaluer la reconnaissance des événements, nous testons notre méthode avec la Section S3 de PETS. Cette base de données représente 6 classes d'événements au sein de la foule: marche, course, formation (fusion), division, évacuation et dispersion. Nous divisons aléatoirement cette base de données (75%) pour l'apprentissage et (25%) pour les tests. Ensuite, suivant la stratégie de un-contre-un, on obtient (99,54%) pour la précision de la classification. En outre, en s'appuyant sur la stratégie de un-contre-reste, nous calculons la précision de la classification sur l'ensemble des tests pour chaque classe séparément, voir le tableau B.6. Comme il est indiqué dans ce tableau, on obtient de bons résultats pour

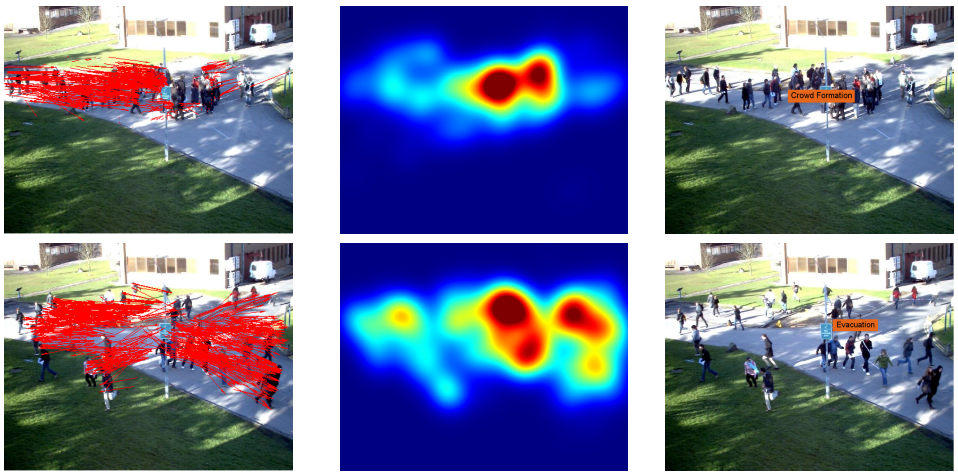
Événements	Marche	Course	Division	Dispersion	Evacuation	Formation
précision (%)	99.41	99.21	100.00	99.87	99.80	99.54

Table B.6: Précision de la classification de méthode pour la reconnaissance des événements sur PETS. S3 suivant la stratégie de un-contre-reste

tous les événements de la foule, y compris la formation et la division de la foule, ce qui justifie la pertinence des attributs proposés.

En suivant quelques mesures de la foule (méthode non supervisée), nous sommes capables aussi de surveiller ce qui se passe dans la scène afin de localiser l'événement et avoir une idée claire sur la densité de personnes participant à chaque événement. Figure B.11 illustre quelques exemples concrets sur la caractérisation avec PETS.

Dans la première ligne de cette figure, un exemple d'événement de formation de la foule est visualisé. Cet événement se caractérise par des personnes venant de directions différentes et qui se déplacent vers le même endroit (comme il est représenté dans la première colonne, indiquant la direction de vecteurs de mouvement). En outre, cet événement



a) vecteurs de mouvement (b) carte de densité (c) l'événement reconnu

Figure B.11: Résultats de la caractérisation des événements de PETS: exemples de formation de la foule et d'évacuation.

se distingue par une baisse du taux de la zone de mouvement dans le temps (égal à 40,72 % dans cette image). Dans la deuxième colonne, nous montrons la carte de densité qui permet de localiser le lieu là où la foule est formée. On constate aussi que la zone de régions denses augmente dans le temps; elle atteint 6,10 % à cette image. Comme il est indiqué dans la troisième colonne, grâce à toutes ces caractéristiques, un événement de formation de la foule est reconnu et localisé.

Dans la deuxième ligne, on illustre un exemple d'évacuation. Comme on peut le voir dans la première colonne, cet événement se caractérise par la divergence des vecteurs de mouvement, car les individus s'éloignent les uns des autres dans des directions différentes. De plus, cet événement se caractérise par une augmentation soudaine de la vitesse: la moyenne de longueur des vecteurs de mouvement à cette image est égale à 12,48 pixels. L'évacuation se distingue également par une augmentation dans le rapport de la zone en mouvement (53,79 %) et une diminution dans le temps des zones denses (comme il est montré dans la deuxième colonne).

B.3.3 Amélioration de la compatibilité entre la vie privée et la surveillance

Dans cette Section, nous nous proposons d'appliquer la mesure de densité de la foule décrite dans la Section B.2.3 dans le contexte de la vie privée en ajustant le niveau de protection de la vie privée selon les besoins locaux. La densité de la foule est sélectionnée en tant que critère pour la protection de la vie privée pour les raisons suivantes: les foules doivent être surveillées en permanence car elles représentent des situations propices où des dangers peuvent se produire tels que les crimes ou les heurts violents. En même temps, les personnes constituant une foule exposent moins d'informations aux agents assurant la

vidéo surveillance. Ainsi, ces personnes ne doivent pas être filtrées de la même manière qu'un individu isolé et entièrement visible.

Pour ne pas réduire la visibilité des informations potentiellement importantes, nous limitons l'application des filtres de préservation de la vie privée à certaines régions d'intérêt, c'est-à-dire seulement les régions qui comportent des renseignements personnels. Dans ce travail, nous considérons la tête comme la partie la plus visible d'un être humain dans une foule. Cependant, dès qu'une personne quitte la foule, elle est perçue comme un sujet isolé. En conséquence, des informations personnelles telles que les vêtements ou la couleur de la peau doivent être cachées. Le diagramme des filtres contextualisés de protection de la vie privée proposés est illustré dans la figure B.12.

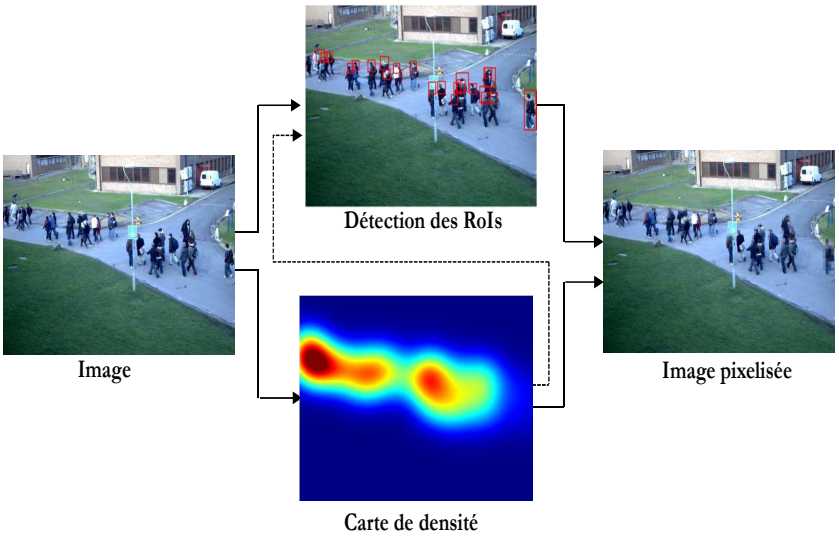


Figure B.12: Organigramme des filtres contextualisés de préservation de la vie privée en utilisant une image de PETS [41], la ligne en pointillés sur cette figure montre que la carte de densité de la foule est également utilisée pour améliorer la détection des personnes

Tout d'abord, pour la détection des régions d'intérêts, nous employons l'extension de détecteur dans des scènes denses décrite dans la Section B.3.1 (ligne en pointillés sur la figure B.12). Ensuite, pour le floutage des personnes, nous appliquons les filtres de préservation de la vie privée à la partie de la tête ou du corps selon que la cible soit isolée ou dans la foule. Enfin, le niveau de protection de la vie privée est adapté en fonction de la densité de la foule. Étant donné un ensemble de paramètres de filtres représentant différents niveaux d'obscurcissement $P = \{P_{min}, ..., P_{max}\}$, pour chaque détection d_j^k, sa valeur moyenne de la densité $\hat{C}_k(d_j^k)$ est utilisée pour choisir le paramètre du filtre.

Comme la visibilité d'une personne dans la scène est également sensible à la distance qui la sépare de la caméra en raison des effets de perspective, nous utilisons cette variable comme un second paramètre pour choisir le niveau d'obscurcissement approprié. Pour estimer la distance, nous adoptons une méthode simple qui consiste à utiliser la taille de

la zone détectée. Néanmoins, étant donné que cette information pourrait être erronée, on a choisi une meilleure méthode qui consiste à calculer le rapport d'aspect et la hauteur perçue à partir de toutes les détections acceptées (cette information peut être obtenue à partir de l'étape de détection). En utilisant cette méthode, nous pourrons prévoir la hauteur $\widetilde{h}_j^k$ et le ratio γ_{k-1} d'une détection. Ainsi, la taille estimée de d_j^k est $\widetilde{S}_j^k = (\widetilde{h}_j^k)^2 * \gamma_{k-1}$ qui est plus robuste que $w_j^k * h_j^k$.

Dans ce travail, nous utilisons deux filtres typiques de protection de la vie privée qui sont:

B.3.3.1 Flou gaussien

Le flou gaussien réside essentiellement en la suppression d'informations dans une région d'intérêt en appliquant un filtrage passe-bas:

$$I_{blur}^k(x,y) = I_k(x,y) * \frac{1}{2\pi {\sigma_{k,j}}^2} e^{\frac{(x^2+y^2)}{2{\sigma_{k,j}}^2}} \tag{B.20}$$

La bande passante $\sigma_{k,j}$ de la gaussienne est adaptée en fonction du niveau de densité de la foule et de la taille estimée.

B.3.3.2 Pixellisation

Ce filtre est basé sur la diminution de la résolution de toutes les régions d'intérêt en remplaçant chaque bloc de pixels par sa moyenne:

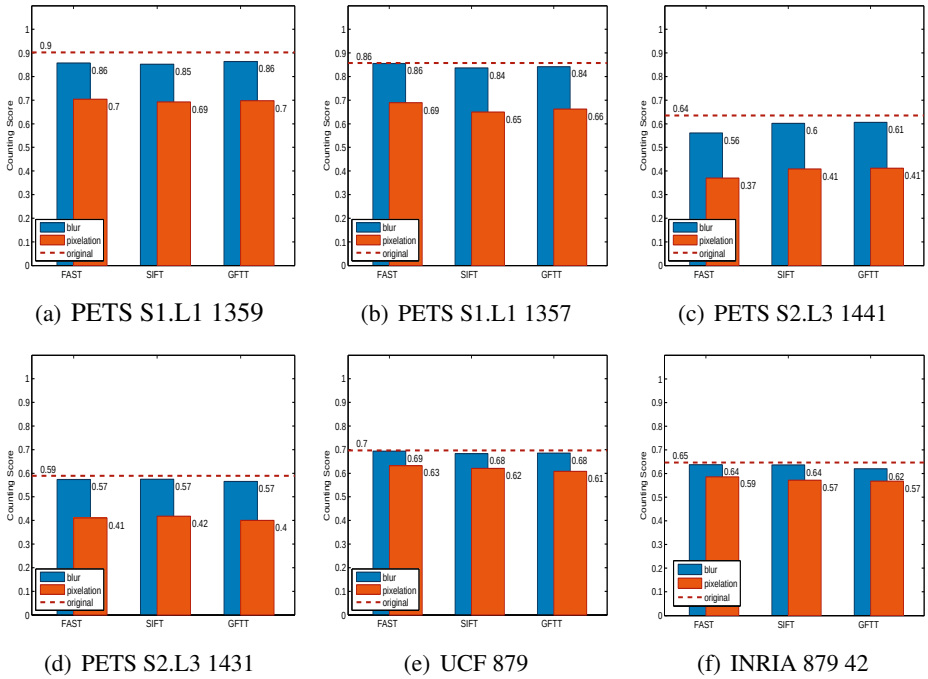
$$I_{pix}^k(x,y) = \frac{1}{b_{k,j}^2} \sum_{i=0}^{b_{k,j}-1} \sum_{j=0}^{b_{k,j}-1} I\left(\left\lfloor \frac{x}{b_{k,j}} \right\rfloor + i, \left\lfloor \frac{y}{b_{k,j}} \right\rfloor + j\right) \tag{B.21}$$

La taille de filtre $b_{k,j} \propto (\hat{C}_k(d_j^k), \widetilde{S}_j^k)$.

B.3.3.3 Résultats expérimentaux

Les filtres de protection de la vie privée proposés sont testés avec des scènes denses de PETS [41], UCF [4] et Data Driven Crowd Analysis [103]. En ce qui concerne l'évaluation, nous adoptons un schéma d'évaluation objective. D'un côté, nous modélisons l'impact des filtres sur l'intelligibilité en évaluant les performances de comptage de personnes avant et après l'application des filtres. Pour ce choix, nous nous appuyons sur le fait que les vidéos filtrées doivent au moins conserver des caractéristiques visuelles pour effectuer des tâches de surveillance telles que la détection de personnes et le comptage. De l'autre côté, nous modélisons la protection de la vie privée par le score inverse d'un algorithme d'appariement basé sur les caractéristiques locales. Un tel algorithme tente d'identifier un individu parmi d'autres à travers l'extraction et la corrélation de caractéristiques locales. Cet algorithme

représente une étape commune à d'autres tâches comme la ré-identification, la reconnaissance ou le suivi, qui pourraient révéler des informations sur l'identité d'une personne. En se basant sur cette hypothèse, un filtre de préservation de la vie privée approprié doit empêcher l'algorithme d'appariement de détecter et de relier correctement les caractéristiques locales.



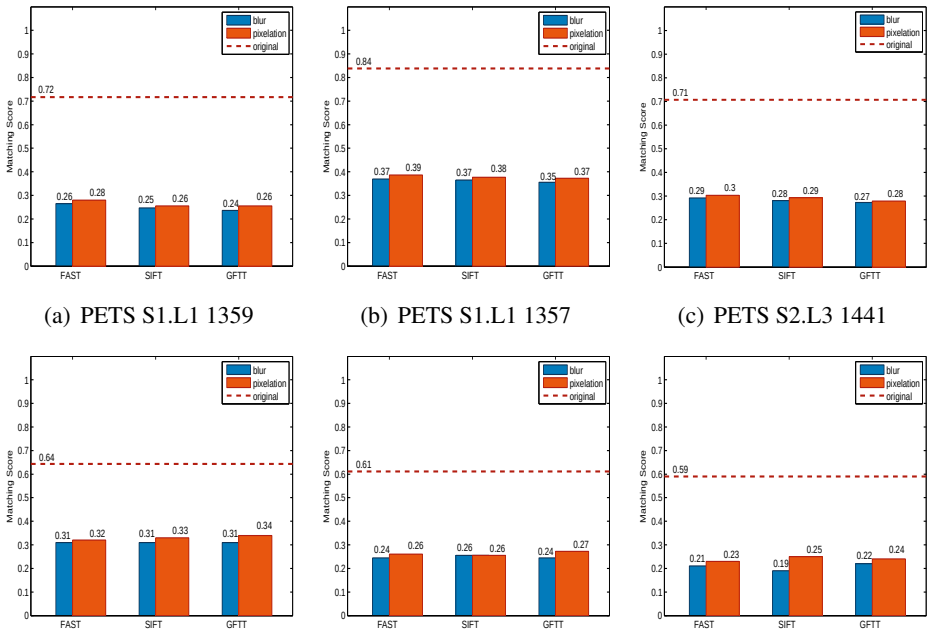
(a) PETS S1.L1 1359 (b) PETS S1.L1 1357 (c) PETS S2.L3 1441

(d) PETS S2.L3 1431 (e) UCF 879 (f) INRIA 879 42

Figure B.13: Scores de comptage sur les séquences filtrées par le flou et la pixellisation, comparés aux résultats originaux

Figure B.13 montre les résultats de comptage en appliquant les filtres de flou gaussien et de pixellisation. Le score d'évaluation est choisi comme le pourcentage $p \in [0, 1]$ des individus détectés correctement par rapport à ceux annotés dans la vérité terrain. La ligne horizontale rouge représente le score de comptage quand aucun filtre de protection n'est appliqué. On peut observer que les résultats de comptage ne diminuent pas de manière significative après l'application des filtres de protection. En moyenne, la baisse du score est de 0,10. Par conséquent, nous sommes toujours en mesure d'effectuer correctement le comptage de personnes avec une marge d'erreur de 10 %. Nous remarquons aussi que les résultats de comptage via le flou gaussien sont mieux par rapport au pixellisation.

Les résultats d'appariement sont présentés sur la figure B.14. Nous pouvons clairement observer une baisse notable des performances de l'algorithme. En moyenne, la baisse est de 0,41. Ces résultats confirment que notre approche de la protection de la vie privée est conforme aux exigences en termes d'intelligibilité et de préservation de la vie privée. Nos filtres génèrent une perte relativement faible pour les scores de comptage de personnes, et



(a) PETS S1.L1 1359 (b) PETS S1.L1 1357 (c) PETS S2.L3 1441

(d) PETS S2.L3 1431 (e) UCF 879 (f) INRIA 879 42

Figure B.14: Scores d'appariement sur les séquences filtrées par le flou et par la pixelisation, comparés aux résultats originaux

donc d'intelligibilité, par rapport à la baisse des performances de l'appariement, et donc le gain au niveau de la protection de la vie privée. Nous remarquons ainsi que pour les deux algorithmes de comptage et d'appariement, les filtres du flou fournissent de meilleurs résultats par rapport à la pixellisation.

B.4 Conclusion

Dans cette thèse,nous avons étudié de nouvelles méthodes pour l'analyse de densité de la foule. Plus précisément, notre contribution dans ce domaine couvre différents aspects, notamment: l'amélioration de l'estimation de densité de la foule par rapport aux méthodes existantes, l'extension de cette estimation du niveau global au niveau pixel en utilisant le suivi de points d'intérêt, l'amélioration de la détection et du suivi des personnes dans les scènes denses en utilisant une estimation préalable de la densité locale, l'utilisation de densité de la foule dans un contexte de la vie privée pour étayer l'équilibre entre la surveillance et la protection de la vie privée, et l'analyse du comportement de la foule.

Cette section résume nos contributions: Dans la première partie de la thèse, nous avons abordé les problèmes liés à l'estimation de la densité de la foule. Afin d'éviter les difficultés typiques rencontrés lors de l'application de la détection et du suivi dans les scènes denses, nos approches traitent des caractéristiques locales au lieu des individus.

Dans la Section B.2.1, nous avons abordé le problème de comptage des personnes. Nous avons consolidé notre contribution à travers deux approches différentes. Nous avons réalisé certaines améliorations dans ce domaine par rapport aux méthodes existantes dans la formulation du problème comme dans les résultats obtenus. Outre le problème des distorsions de perspective, nous avons traité le problème des variations de densité par la formulation d'une nouvelle fonction de poids basée sur la densité des caractéristiques locales. En comparaison avec d'autres méthodes, les résultats ont démontré que nos approches se distinguent par une bonne précision de comptage dans des situations ayant des occultations et des distorsions de perspective.

Dans la Section B.2.2, nous avons abordé le problème de l'estimation du niveau de la foule qui est une autre représentation de la densité. En particulier, nous nous sommes focalisés sur l'analyse de texture pour caractériser la densité au niveau de région. Ensuite, nous avons appliqué PCA et LDA pour améliorer la puissance discriminante et descriptive de LBP. En outre, nous avons inclus une étude comparative afin de prouver que parmi de nombreuses caractéristiques de texture, peu sont discriminantes pour la densité. Nous avons également proposé un nouvel algorithme de multi-classes SVM en utilisant des scores de pertinence automatiques. Les résultats expérimentaux ont mis en évidence le rôle de la réduction de dimension de LBP. De plus, l'application de l'algorithme multi-classes SVM a donné de bons résultats en termes de précision de la classification tout en maintenant un coût de calcul faible par rapport aux autres méthodes existantes.

Dans la Section B.2.3, nous avons étendu l'estimation de la densité au niveau local en appliquant l'estimation par noyau sur les positions des caractéristiques locales en mouvement. Les informations de densité ont été représentées comme un modèle statistique de caractéristiques spatio-temporelles. Ce processus comprend une étape de suivi pour atténuer les effets des caractéristiques non pertinentes vis-à-vis de la densité. Notre approche a été testée sur des vidéos de différentes bases de données. En se basant sur notre méthodologie d'évaluation, les résultats démontrent l'efficacité des suivis de caractéristiques pour l'estimation de la densité.

Durant la première partie de la thèse, nos contributions ont été illustrées à travers plusieurs composants d'analyse des vidéos. Ces composants peuvent être classés en trois catégories: (1) L'extraction de caractéristiques visuelles: il s'agit ici de présenter les différentes méthodes que nous avons utilisées pour transformer les données brutes d'une séquence vidéo en une description plus sophistiquée. Cette description illustre différentes propriétés d'un objet ou d'un événement selon le problème à résoudre. Ces caractéristiques varient du dense au clairsemé et du local au global. (2) L'estimation de mouvement: cette étape est fondamentale dans notre travail, puisque notre étude porte généralement sur la foule dynamique. Durant la première partie de la thèse, nous avons appliqué des techniques différentes, à savoir la soustraction de l'arrière-plan par GMM, le flot optique dense par l'algorithme de Farneback, et le flot optique par RLOF. (3) La reconnaissance des formes et l'apprentissage automatique: en utilisant l'apprentissage automatique et les tech-

niques de reconnaissance de formes, l'ensemble des caractéristiques extraites sont classées, partitionnées, et régressées selon la tâche qu'on souhaite réaliser. Précisément, nous avons appliqué la classification par Support Vector Machine, la régression gaussienne, la répartition par DBSCAN et la réduction de la dimension par PCA et LDA.

Dans la deuxième partie de cette thèse, nous avons abordé certains problèmes liés à l'analyse de la foule dans une nouvelle optique. En prenant en compte les difficultés rencontrées lors de l'analyse des vidéos des scènes denses, nous avons utilisé une estimation préalable de la densité de la foule pour compléter certaines applications. En particulier, nous avons utilisé le modèle local de la densité, présenté dans la Section B.2.3, afin d'atteindre les objectifs suivants: Étant donné que la détection et le suivi des personnes au sein des foules sont difficiles, dans la Section B.3.1 nous avons introduit dans le processus de détection des connaissances supplémentaires sur la répartition spatiale des individus à l'aide de la carte de la densité. La combinaison de densité avec la localisation d'individus a été effectuée en utilisant un paramétrage adaptatif. Les résultats de détection améliorés ont été étendus dans un schéma de suivi-par-détection. L'évaluation de notre méthode sur des vidéos de différentes bases de données démontre que notre système obtient des meilleurs résultats que ceux de l'algorithme de base. Cela a abouti à des améliorations substantielles des résultats de suivi.

Ensuite, la Section B.3.2 a été consacrée à la détection de changement et la reconnaissance des événements dans la foule. Notre approche mobilise des caractéristiques de bas niveau en faveur des applications de haut niveau et fait abstraction des étapes intermédiaires comme la détection et le suivi d'objets. Le suivi de la foule repose sur des trajectoires à long terme des caractéristiques locales. Ensuite, la détection de changement de la foule et la reconnaissance des événements ont été effectuées par la modélisation de la densité et de mouvement extraits de ces trajectoires. Les résultats expérimentaux ont montré des bons résultats dans la détection précoce des changements et une reconnaissance précise des événements.

Finalement, nous avons montré dans la Section B.3.3 que l'estimation de la densité peut être utilisée pour ajuster le niveau de la protection de la vie privée. En s'appuyant sur les cartes de densité, le niveau de floutage diminue en fonction de la densité locale. Nous avons proposé des filtres contextualisés de protection de la vie privée qui sont efficaces aussi bien dans des scènes à haute densité que dans des scènes à faible densité. En outre, une évaluation objective en intelligibilité contre la préservation de la vie privée a été proposée. Les résultats expérimentaux ont démontré que nos filtres donnent de bonnes performances sur des tâches d'intelligibilité communes (telles que la détection de personnes et le comptage) tout en protégeant la vie privée de personnes (difficultés pour établir l'appariement).

En résumé, dans la deuxième partie de la thèse, nous avons démontré qu'en pratique, l'estimation de la densité de la foule comporte plusieurs avantages. Bien que cette estimation joue un rôle crucial dans la surveillance de la foule afin de détecter les situations d'encombrement extrême, elle peut être utilisée pour compléter d'autres applications en

vidéosurveillance. Dans cette optique, trois applications différentes ont été étudiées.

Bibliography

[1] U. of minnesota crowd activity dataset. http://www.mha.cs.umn.edu/Movies/Crowd-Activity-All.avi. xiii, 103, 108, 109, 146

[2] G. Acampora, V. Loia, G. Percannella, and M. Vento. Trainable estimators for indirect people counting: A comparative study. In *FUZZ-IEEE*, pages 139–145, 2011. 16

[3] A. Albiol, M. J. Silla, A. Albiol, and J. M. Mossi. Video analysis using corner motion statistics. In *IEEE International Workshop on PETS*, pages 31–37, 2009. 16, 19, 23, 24, 26, 27, 34, 98, 133

[4] S. Ali and M. Shah. A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis. In *CVPR 07*, pages 1–6, 2007. 59, 77, 90, 98, 141, 150

[5] S. Ali and M. Shah. Floor fields for tracking in high density crowd scenes. In *ECCV (2)*, pages 1–14, 2008. 18, 67, 68

[6] A. Badii, M. Einig, M. Tiemann, D. Thiemert, and C. Lallah. Visual context identification for privacy-respecting video analytics. In *Multimedia Signal Processing (MMSP), 2012 IEEE 14th International Workshop on*, pages 366–371, 2012. 86

[7] M. Bauml and R. Stiefelhagen. Evaluation of local features for person re-identification in image sequences. In *AVSS*, pages 291–296, 2011. 91

[8] B.Benfold and I. Reid. Stable multi-target tracking in real-time surveillance video. In *CVPR*, pages 3457–3464, June 2011. 71

[9] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:711–720, 1997. 44

[10] S.S. Blackman. Multiple hypothesis tracking for multiple target tracking. *Aerospace and Electronic Systems Magazine, IEEE*, 19(1):5 –18, Jan. 2004. 75

[11] M. D. Breitenstein, F. Reichlin, B. Leibe, E. K. Meier, and L. V. Gool. Robust tracking-by-detection using a detector confidence particle filter. In *IEEE International Conference on Computer Vision*, October 2010. 17, 18

[12] A. Briassouli and I. Kompatsiaris. Spatiotemporally localized new event detection in crowds. In *ICCV Workshops*, pages 928–933. IEEE, 2011. 19, 97

[13] M. Butenuth, F. Burkert, F. Schmidt, S. Hinz, D. Hartmann, A. Kneidl, A. Borrmann, and B. Sirmacek. Integrating pedestrian simulation, tracking and event detection for crowd analysis. In *ICCV Workshops*, pages 150–157, 2011. 55, 139

[14] A. Cavallaro. Privacy in video surveillance. *IEEE SIGNAL PROCESSING MAGAZINE*, March, 2007. 85

[15] A. B. Chan, Z. S. J. Liang, and N. Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–7, 2008. 16, 23, 24, 25, 26, 27, 34

[16] A. B. Chan, M. Morrow, and N. Vasconcelos. Analysis of crowded scenes using holistic properties. In *IEEE International Workshop on PETS*, 2009. 19

[17] A. B. Chan, M. Morrow, and N. Vasconcelos. Analysis of crowded scenes using holistic properties. In *Proceedings of the 11th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, PETS-09, June 2009. 19, 34, 97

[18] A. B. Chan and N. Vasconcelos. Bayesian poisson regression for crowd counting. In *IEEE International Conference on Computer Vision (ICCV)*, pages 545–551, 2009. 16

[19] D. Y. Chen and P.C. Huang. Motion-based unusual event detection in human crowds. *J. Visual Communication and Image Representation*, 22(2):178–186, 2011. xiii, xvi, 19, 104, 106, 109

[20] D. Clark and B.-N. Vo. Convergence analysis of the gaussian mixture phd filter. In *IEEE Transactions on Signal Processing*, volume 55, pages 1208–1209, April 2007. 76

[21] R. T. Collins, A. J. Lipton, and T. Kanade. Introduction to the special section on video surveillance. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):745–746, 2000. 12

[22] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. pages 142–149, 2000. 13

[23] D. Conte, P. Foggia, G. Percannella, F. Tufano, and M. Vento. A method for counting people in crowded scenes. In *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2010. 16, 23, 24, 26, 27, 28, 34, 35, 133

[24] C. Corinna and V. Vladimir. Support-vector networks. *Machine Learning*, 20, 1995. 44, 136

[25] R. Cutler and L. Davis. Robust real-time periodic motion detection, analysis, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:781–796, 1999. 14

[26] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 886–893, 2005. 42, 48, 69, 88

[27] Angela D'angelo and Jean-Luc Dugelay. People re-identification in camera networks based on probabilistic color histograms. In *3DIP 2011, Electronic Imaging Conference on 3D Image Processing and Applications, Vol. 7882, 23-27 January, 2011, San Francisco, CA, USA*, San Francisco, UNITED STATES, 01 2011. 13

[28] A. C. Davies, J. H. Yin, and S. A. Velastin. Crowd monitoring using image processing. In *Electron. Commun. Eng. J.*, volume 7, pages 37–47, 1995. 16, 23

[29] I. R. de Almeida and C. R. Jung. Change detection in human crowds. In *Conference on Graphics, Patterns and Images*, number 26, 2013. xiii, 19, 100, 101, 104, 106, 108, 109, 145

[30] H. Dee and D. Hogg. Detecting inexplicable behaviour. In *In: Proceedings of the British Machine Vision Conference, The British Machine Vision Association*, pages 477–486, 2004. 97

[31] A. R. Dick and M. J. Brooks. Issues in automated visual surveillance. In Changming Sun, Hugues Talbot, Sebastien Ourselin, and Tony Adriaansen, editors, *DICTA*, pages 195–204. CSIRO Publishing, 2003. 12

[32] F. Dufaux and T. Ebrahimi. Scrambling for privacy protection in video surveillance systems. In *IEEE Trans. on Circ. Syst. for Video Tech.*, volume 18, pages 1168–1174, 2008. 90

[33] H. Edelsbrunner, D. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. In *IEEE Transactions on Information Theory*, volume 29, pages 551–559, 1983. 28

[34] V. Eiselein, D. Arp, M. Pätzold, and T. Sikora. Real-time multi-human tracking using a probability hypothesis density filter and multiple detectors. In *AVSS*, 2012. 17, 76

[35] A. Elgammal, D. Harwood, and L. Davis. Non-parametric model for background subtraction. In *Proceedings of the 6th European Conference on Computer Vision-Part II*, ECCV '00, pages 751–767, London, UK, UK, 2000. Springer-Verlag. 13

[36] Rémi Emonet, J. Varadarajan, and Jean-Marc Odobez. Temporal Analysis of Motif Mixtures using Dirichlet Processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014. 19

[37] M. Ester, H. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. of International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 226–231, 1996. 28

[38] R. H. Evangelio and T. Sikora. Complementary background models for the detection of static and moving objects in crowded environments. In *AVSS*, 2011. 17

[39] G. Farneback. Two-frame motion estimation based on polynomial expansion. In *Proc. of 13th Scandinavian Conference on Image Analysis*, pages 363–370, 2003. 28, 31, 119, 131

[40] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010. xii, xv, 18, 68, 69, 70, 77, 82, 88, 142, 143

[41] J. Ferryman and A. Shahrokni. Pets2009: Dataset and challenge. In *PETS*, pages 1–6, 2009. xii, xiii, xiv, 32, 46, 59, 72, 77, 88, 90, 133, 137, 141, 149, 150

[42] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. P.: Multi-camera people tracking with a probabilistic occupancy map. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007. 18

[43] Gian Luca Foresti, Petri Mahonen, and Carlo S. Regazzoni, editors. *Multimedia Video-Based Surveillance Systems: Requirements, Issues and Solutions*. Kluwer Academic Publishers, Norwell, MA, USA, 2000. 12

[44] H. Fradi and J. L. Dugelay. Low level crowd analysis using frame-wise normalized feature for people counting. In *IEEE International Workshop on Information Forensics and Security*, December 2012. 29

[45] H. Fradi and J. L. Dugelay. People counting system in crowded scenes based on feature regression. In *EUSIPCO 2012, European Signal Processing Conference*, August 2012. 26

[46] H. Fradi and J. L. Dugelay. Robust foreground segmentation using improved gaussian mixture model and optical flow. In *International Conference on Informatics, Electronics and Vision*, 2012. 29, 31, 35, 132, 134

[47] C. Garate, P. Bilinski, and F. Bremond. Crowd Event Recognition using HOG Tracker. In *Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-Winter)*, pages 1–6. IEEE, December 2009. 19, 98, 104

[48] G. Gennari and G. D. Hager. Probabilistic data association methods in visual tracking of groups. In *in Proc. CVPR, 2004*, pages 1063–1069, 2004. 18

[49] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester. Discriminatively trained deformable part models, release 5. http://people.cs.uchicago.edu/ rbg/latent-release5/. 70

[50] I. Haritaoglu, D. Harwood, and L. S. Davis. W4: Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:809–830, 2000. 13

[51] E. Hayman and J. O. Eklundh. Statistical background subtraction for a mobile observer. In *International Conference on Pattern Recognition*, 2003. 13

[52] M. Hofmann, M. Haag, and G. Rigoll. Unified hierarchical multi-object tracking using global data association. In *PETS*, 2013. 17

[53] D. Hoiem, A. A. Efros, and M. Hebert. Putting objects in perspective. *International Journal of Computer Vision*, 80(1):3–15, 2008. 72, 143

[54] Y. Hou and G. Pang. Automated people counting at a mass site. In *IEEE International Conference on Automation and Logistics*, pages 464–469, 2008. 24

[55] Y. L. Hou and G. K. H. Pang. People counting and human detection in a challenging situation. In *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, volume 41, pages 24–33, 2011. 18, 68

[56] C. W. Hsu and C. J. Lin. A comparison of methods for multiclass support vector machines. In *IEEE Trans. Neural Networks*, volume 13, pages 415–425, 2002. 44, 48

[57] H. Idrees, I. Saleemi, and M. Shah. Multi-source multi-scale counting in extremely dense crowd images. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, 2013. 113

[58] N. Ihaddadene and C. Djeraba. Real-time crowd motion analysis. In *ICPR*, pages 1–4. IEEE, 2008. 19, 97

[59] M. Isard and J. MacCormick. Bramble: A bayesian multiple-blob tracker. In *ICCV*, pages 34–41, 2001. 13

[60] J.C.S. Jacques Junior, S. Raupp Musse, and C.R. Jung. Crowd analysis using computer vision techniques. In *IEEE Signal Processing Magazine*, volume 27, pages 66–77, 2010. 15

[61] I. T. Jolliffe. Principal component analysis. *2nd ed. New-York: Springer-Verlag*, 2002. 43

[62] P. Kaewtrakulpong and R. Bowden. An improved adaptive background mixture model for realtime tracking with shadow detection. In *2nd European Workshop on Advanced Video Based Surveillance Systems*, 2001. 118

[63] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960. 75

[64] V. Kaltsa, A. Briassouli, I. Kompatsiaris, and M. G. Strintzis. Timely, robust crowd event characterization. In *ICIP*, pages 2697–2700, 2012. 19, 105, 146

[65] K. Keung, L. Y. Xu, and X. Wu. Crowd density estimation using texture analysis and learning. In *IEEE International Conference on Robotics and Biomimetics*, pages 214–219, 2006. 17, 39

[66] S. M. Khan and M. Shah. A multiview approach to tracking people in crowded scenes using a planar homography constraint. In *In European Conference on Computer Vision*, 2006. 18

[67] P. Kilambi, O. Masoud, and N. Papanikolopoulos. Crowd analysis at mass transit site. In *Proc. IEEE Intell. Transp. Syst. Conf.*, pages 753–758, 2006. 16, 23

[68] L. Kratz and K. Nishino. Tracking with local spatio-temporal motion patterns in extremely crowded scenes. In *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 693–700, 2010. 67

[69] Ulrich H.-G. Kressel. Pairwise classification and support vector machines. In *Advances in kernel methods: support vector learning, MIT Press*, pages 255–268, 1999. 44

[70] B. Leibe, K. Schindler, and L. V. Gool. Coupled detection and trajectory estimation for multi-object tracking. In *In ICCV*, pages 1–8, 2007. 18

[71] V. Lempitsky and A. Zisserman. Learning to count objects in images. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 1324–1332. 2010. 58, 59

[72] L. Li, W. Huang, I. Y. H. Gu, and Q. Tian. Foreground object detection from videos containing complex background. In *ACM MM*, 2003. xiii, 121, 122, 123

[73] M. Li, Z. Zhang, K. Huang, and T. Tan. Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection. In *ICPR*, pages 1–4, 2008. 16

[74] S.F. Lin, J.Y. Chen, and H.X. Chao. Estimation of number of people in crowded scenes using perspective transformation. In *IEEE Trans. System, Man, and Cybernetics*, volume 31, pages 645–654, 2001. 16

[75] W.-C. Lin and Y. Liu. Tracking dynamic near-regular texture under occlusion and rapid movements. In Ales Leonardis, Horst Bischof, and Axel Pinz, editors, *ECCV (2)*, volume 3952 of *Lecture Notes in Computer Science*, pages 44–55. Springer, 2006. 18

[76] David G. Lowe. Distinctive image features from scale-invariant keypoints. In *Int. J. Comput. Vision*, pages 91–110, 2004. 27, 55, 59, 131, 140

[77] R. Ma, L. Li, W. Huang, and Qi Tian. On pixel count based crowd density estimation for visual surveillance. In *IEEE Conference on Cybernetics and Intelligent Systems*, pages 170–173, 2004. 24

[78] W. Ma, L. Huang, and C. Liu. Advanced local binary pattern descriptors for crowd estimation. *Computational Intelligence and Industrial Application*, 2:958–962, 2008. 17, 40

[79] W. Ma, L. Huang, and C. Liu. Crowd density analysis using co-occurrence texture features. *International Conference on Computer Sciences and Convergence Information Technology*, pages 170–175, 2010. 40, 41

[80] W. Ma, L. Huang, and Ch. Liu. Estimation of crowd density using image processing. *Computer Sciences and Convergence Information Technology*, pages 170–175, 2010. 17, 39

[81] R. P. S. Mahler. *Statistical Multisource-Multitarget Information Fusion*. Artech House, Inc., Norwood, MA, USA, 2007. 75

[82] R.P.S. Mahler. Multitarget bayes filtering via first-order multitarget moments. *Aerospace and Electronic Systems, IEEE Transactions on*, 39(4):1152 – 1178, oct. 2003. 75

[83] A. N. Marana, S. A. VelaStin, L. F. Costa, and R. A. Lotufo. Estimation of crowd density using image processing. *IEEE Colloquium Image Processing for Security Applications*, 11:1–8, 1997. 16, 17, 23, 39, 48

[84] A. N. Marana and V. V. Verona. Wavelet packet analysis for crowd density estimation. In *Proceedings of the IASTED International Symposia on Applied Informatics*, pages 535–540, 2001. 17, 39

[85] V.Y. Mariano, J. Min, J.-H. Park, R. Kasturi, D. Mihalcik, D. Doermann, and T. Drayer. Performance Evaluation of Object Detection Algorithms. In *ICPR*, pages 965–969, 2002. 77

[86] R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behavior detection using social force model. In *CVPR*, pages 935–942, 2009. xiii, 19, 97, 106, 109

[87] S. Moncrieff, S. Venkatesh, and G. West. Context aware privacy in visual surveillance. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4, 2008. 86

[88] S. M. Mousavi, S. O. Shahdi, and S. A. R. Abu-Bakar. Crowd estimation using histogram model classification based on improved uniform local binary pattern. *International Journal of Computer and Electrical Engineering*, 4:256–259, 2012. 17, 40, 43

[89] M. Muja and D. G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *In VISAPP International Conference on Computer Vision Theory and Applications*, pages 331–340, 2009. 91

[90] V. Norris, M. McCahill, and D. Wood. Editorial: The growth of CCTV: a global perspective on the international diffusion of video surveillance in publicly accessible space. *Surveillance and Society*, 2(2/3):110–135, 2004. 85

[91] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002. 17, 40, 41

[92] N. Paragios and V. Ramesh. A mrf-based approach for real-time subway monitoring. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 1034–1040, 2001. 16, 24

[93] M. Pätzold and T. Sikora. Real-time person counting by propagating networks flows. In *AVSS*, pages 66–70, 2011. 17

[94] A. Polus, J. L. Schofer, and A. Ushpiz. Pedestrian flow and level of service. *Journal of Transportation. Engineering*, 109:46–56, 1983. 17, 39, 40, 41, 47, 100, 134

[95] A. Prati, I. Mikic, M. Trivedi, and R. Cucchiara. Detecting moving shadows: Algorithms and evaluation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, number 25, pages 918–923, 2003. 119

[96] C. E. Rasmussen and C. K. I. Williams. Gaussian Processes for Machine Learning. In *The MIT Press*, December 2006. 32, 133

[97] M. Redi and B. Mérialdo. A multimedia retrieval framework based on automatic graded relevance judgments. In *18th International Conference on Multimedia Modeling (MMM)*, 2012. 45

[98] Carlo S. Regazzoni and Alessandra Tesei. Distributed data fusion for real-time crowding estimation. *Signal Processing*, 53(1):47–63, August 1996. 16

[99] B. Ristic, B.-N. Vo, D. Clark, and B.-T. Vo. A metric for performance evaluation of multi-target tracking algorithms. *IEEE Transactions on Signal Processing*, 59(7):3452–3457, 2011. 78, 144

[100] M. Rodriguez, S. Ali, and T. Kanade. Tracking in unstructured crowded scenes. In *ICCV*, pages 1389–1396, 2009. 18, 67, 68

[101] M. Rodriguez, I. Laptev, J. Sivic, and J.-Y. Audibert. Density-aware person detection and tracking in crowds. In *ICCV*, pages 2423–2430, 2011. 18, 68, 69, 70

[102] M. Rodriguez, J. Sivic, and I. Laptev. Analysis of crowded scenes in video. In J. Y. Doufour, editor, *Intelligent Video Surveillance Systems*, pages 251–272. Wiley, 2012. 15

[103] M. Rodriguez, J. Sivic, I. Laptev, and J.-Y. Audibert. Data-driven crowd analysis in videos. In *ICCV*, 2011. 59, 77, 90, 141, 150

[104] E. Rosten, R. Porter, and T. Drummond. Faster and better: A machine learning approach to corner detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 32:105–119, 2010. 31, 55, 139

[105] J. Sun M. Vicencio-Silva D. Aubert A. Lemmer P. Brice L. Khoudor S. Velastin, M. Sanchez-Svensson and S. Kallweit. D7p: Innovative tools for security in transports. *Technical Report GRD1-2000-10601, PRISMATICA Project, 5th Framework Programme*, 2003. 11

[106] S. Saxena, F. Brémond, M. Thonnat, and R. Ma. Crowd behavior recognition for video surveillance. In *Proceedings of the 10th International Conference on Advanced Concepts for Intelligent Vision Systems*, ACIVS '08, pages 970–981, Berlin, Heidelberg, 2008. Springer-Verlag. 19

[107] D. Schuhmacher, B.-T. Vo, and B.-N. Vo. A consistent metric for performance evaluation of multi-object filters. *IEEE Trans. Signal Processing*, 56(8):3447–3457, 2008. 78

[108] A. Senior, A. Hampapur, Y.-L. Tian, L. Brown, S. Pankanti, and R. Bolle. Appearance models for occlusion handling. In *in: International Workshop on Performance Evaluation of Tracking and Surveillance*, 2001. 13

[109] A. Senior, S. Pankanti, A. Hampapur, L. Brown, Ying-Li Tian, A. Ekin, J. Connell, Chiao-Fe Shu, and Max Lu. Enabling video privacy through computer vision. *Security Privacy, IEEE*, 3(3):50–57, 2005. 85

[110] T. Senst, V. Eiselein, R. H. Evangelio, and T. Sikora. Robust modified l2 local optical flow estimation and feature tracking. In *IEEE Workshop on Motion and Video Computing (WMVC)*, pages 685–690, January 2011. 56

[111] T. Senst, V. Eiselein, and T. Sikora. Robust local optical flow for feature tracking. *Transactions on Circuits and Systems for Video Technology*, 09(99), 2012. 54, 56, 140

[112] D.N. Serpanos and A. Papalambrou. Security and privacy in distributed smart cameras. *Proceedings of the IEEE*, 96(10):1678 –1687, oct. 2008. 85

[113] M. Shah, O. Javed, and K. Shafique. Automated visual surveillance in realistic scenarios. *IEEE MultiMedia*, 14(1):30–39, 2007. 12

[114] Mubarak Shah. Visual crowd surveillance is like hydrodynamics. In *ACM Multimedia*, pages 3–4, 2010. 15

[115] Y. B. Shalom and E. Tse. Tracking in a cluttered environment with probabilistic data association. *Automatica*, 11(5):451 – 460, 1975. 75

[116] S. Shan, W. Gao, Y. Chang, B. Cao, and P. Yang. Review the strength of gabor features for face recognition from the angle of its robustness to mis-alignment. *International Conference on Pattern Recognition*, 2004. 48

[117] J. Shi and C. Tomasi. Good features to track. In *CVPR*, pages 593 –600, 1994. 55, 59, 140

[118] Y. Shi, Y. Gao, and R. Wang. Real-time abnormal event detection in complicated scenes. In *Proceedings of the 2010 20th International Conference on Pattern Recognition*, ICPR '10, pages 3653–3656, Washington, DC, USA, 2010. IEEE Computer Society. 19

[119] S. Srivastava, K. K. Ng, and E. J. Delp. Crowd flow estimation using multiple visual features for scenes with changing crowd densities. In *IEEE International Conference on Advanced Video and Signal-Based Surveillance*, pages 60–65, 2011. 24, 25

[120] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 246–252, 1999. 13, 30, 117

[121] R. Stiefelhagen, K. Bernardin, R. Bowers, J. Garofolo, D. Mostefa, and P. Soundararajan. The clear 2006 evaluation. In *Multimodal Technologies for Perception of Humans*, volume 4122, pages 1–44, 2007. 77, 143

[122] Y. Tian, R. Feris, and A. Hampapur. Real-time detection of abandoned and removed objects in complex environments. In *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. 14

[123] C. Tomasi and T. Kanade. Detection and tracking of point features. Technical report CMU-CS-91-132, CMU, 1991. 56

[124] Tsai and Y. Roger. An efficient and accurate camera calibration technique for 3d machine vision. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 364–374, 1986. 41

[125] Bala Subburaman Venkatesh, Adrien Descamps, and Cyril Carincotte. Counting people in the crowd using a generic head detector. In *AVSS*, pages 470–475. IEEE Computer Society, 2012. 16

[126] B.-N. Vo and W.-K. Ma. The gaussian mixture probability hypothesis density filter. *Signal Processing, IEEE Transactions on*, 54(11):4091 –4104, nov. 2006. 75, 144

[127] A. W. *Protecting Privacy in Video Surveillance*. Springer, 2009. 12

[128] Z. Wang, H. Liu, Y. Qian, and T. Xu. Crowd density estimation based on local binary pattern co-occurrence matrix. In *Proceedings of the 2012 IEEE International Conference on Multimedia and Expo Workshops*, pages 372–377, 2012. 17, 40

[129] T. Winkler and B. Rinner. A systematic approach towards user-centric privacy and security for smart camera networks. In *Proceedings of the International Conference on Distributed Smart Cameras (ICDSC) 2010*, 2010. 85

[130] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *ICCV*, volume 1, pages 90–97, 2005. 16

[131] C. R. Huang Y. T. Chen, C. S. Chen and Y. P. Hung. Efficient hierarchical method for background subtraction. In *International Conference on Pattern Recognition*, volume 40, pages 2706–2715, 2007. 13

[132] H. Yang, H. Su, S. Zheng, S. Wei, and Y. Fan. The large-scale crowd density estimation based on sparse spatiotemporal local binary pattern. *IEEE International Conference on Multimedia and Expo*, pages 1–6, 2011. 17, 40

[133] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Comput. Surv.*, 38(4), December 2006. 13

[134] B. Zhan, D. N. Monekosso, P. Remagnino, S. A. Velastin, and L.-Q. Xu. Crowd analysis: a survey. *Mach. Vision Appl.*, 19(5-6):345–357, September 2008. 15

[135] Y. Zhang, L. Qiny, H. Yao, P. Xu, and Q. Huang. Beyond particle flow: Bag of trajectory graphs for dense crowd event recognition. In *ICIP*, 2013. 19

[136] Tao Zhao, Ram Nevatia, and Bo Wu. Segmentation and tracking of multiple humans in crowded environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(7):1198–1211, 2008. 18

[137] X. Zhao, D. Gong, and G. Medioni. Tracking using motion patterns for very crowded scenes. In *ECCV*, pages 315–328, 2012. 67

[138] Z. Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *ICPR*, number 2, pages 28–31, 2004. xiii, 31, 35, 59, 119, 121, 122, 123, 133

[139] Z. Zivkovic and F. van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recogn. Lett.*, 27(7):773–780, May 2006. 13