VISUAL ANALYSIS OF EXTREMELY DENSE CROWDED SCENES

by

HAROON IDREES
B.Sc (Hons) Lahore University of Management Sciences, 2007

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Fall Term
2014

Major Professor: Mubarak Shah

ABSTRACT

Visual analysis of dense crowds is particularly challenging due to large number of individuals, occlusions, clutter, and fewer pixels per person which rarely occur in ordinary surveillance scenarios. This dissertation aims to address these challenges in images and videos of extremely dense crowds containing hundreds to thousands of humans. The goal is to tackle the fundamental problems of counting, detecting and tracking people in such images and videos using visual and contextual cues that are automatically derived from the crowded scenes.

For counting in an image of extremely dense crowd, we propose to leverage multiple sources of information to compute an estimate of the number of individuals present in the image. Our approach relies on sources such as low confidence head detections, repetition of texture elements (using SIFT), and frequency-domain analysis to estimate counts, along with confidence associated with observing individuals, in an image region. Furthermore, we employ a global consistency constraint on counts using Markov Random Field which caters for disparity in counts in local neighborhoods and across scales. We tested this approach on crowd images with the head counts ranging from 94 to 4543 and obtained encouraging results. Through this approach, we are able to count people in images of high-density crowds unlike previous methods which are only applicable to videos of low to medium density crowded scenes. However, the counting procedure just outputs a single number for a large patch or an entire image. With just the counts, it becomes difficult to measure the counting error for a query image with unknown number of people. For this, we propose to localize humans by finding repetitive patterns in the crowd image. Starting with detections from an underlying head detector, we correlate them within the image after their selection through several criteria: in a pre-defined grid, locally, or at multiple scales by automatically finding the patches that are most representative of recurring patterns in the crowd image. Finally, the set of generated hypotheses is selected using binary integer quadratic programming with Special Ordered Set (SOS) Type 1 constraints.

Human Detection is another important problem in the analysis of crowded scenes where the goal is to place a bounding box on visible parts of individuals. Primarily applicable to images depicting medium to high density crowds containing several hundred humans, it is a crucial prerequisite for many other visual tasks, such as tracking, action recognition or detection of anomalous behaviors, exhibited by individuals in a dense crowd. For detecting humans, we explore context in dense crowds in the form of locally-consistent scale prior which captures the similarity in scale in local neighborhoods with smooth variation over the image. Using the scale and confidence of detections obtained from an underlying human detector, we infer scale and confidence priors using Markov Random Field. In an iterative mechanism, the confidences of detections are modified to reflect consistency with the inferred priors, and the priors are updated based on the new detections. The final set of detections obtained are then reasoned for occlusion using Binary Integer Programming where overlaps and relations between parts of individuals are encoded as linear constraints. Both human detection and occlusion reasoning in this approach are solved with local neighbor-dependent constraints, thereby respecting the inter-dependence between individuals characteristic to dense crowd analysis. In addition, we propose a mechanism to detect different combinations of body parts without requiring annotations for individual combinations.

Once human detection and localization is performed, we then use it for tracking people in dense crowds. Similar to the use of context as scale prior for human detection, we exploit it in the form of motion concurrence for tracking individuals in dense crowds. The proposed method for tracking provides an alternative and complementary approach to methods that require modeling of crowd flow. Simultaneously, it is less likely to fail in the case of dynamic crowd flows and anomalies by minimally relying on previous frames. The approach begins with the automatic identification of prominent individuals from the crowd that are easy to track. Then, we use Neighborhood Motion Concurrence to model the behavior of individuals in a dense crowd, this predicts the position of an individual based on the motion of its neighbors. When the individual moves with the crowd flow, we use Neighborhood Motion Concurrence to predict motion while leveraging five-frame instantaneous flow in case of dynamically changing flow and anomalies. All these aspects are then embedded in a framework which imposes hierarchy on the order in which positions of individuals are updated. The results are reported on eight sequences of medium to high density crowds and our approach performs on par with existing approaches without learning or modeling patterns of crowd flow.

We experimentally demonstrate the efficacy and reliability of our algorithms by quantifying the performance of counting, localization, as well as human detection and tracking on new and challenging datasets containing hundreds to thousands of humans in a given scene.

To my family,

Maa, Unaiza and Arsalan

ACKNOWLEDGMENTS

First and foremost, I would thank Dr. Mubarak Shah for his guidance and support. He has been a source of motivation and inspiration, and this dissertation would not have been possible without him. I am grateful to my colleagues at Center For Research in Computer Vision at UCF for their advice and company especially Imran Saleemi with whom I worked jointly on several projects. I would also thank Dr. Kenneth Stanley, Dr. George Atia, Dr. Neils da Vitoria Lobo and Dr. Bahaa Saleh for serving on my committee. Finally, I am grateful and indebted to my mother for her encouragement and support, and to my late father for instilling an appreciation of knowledge and reasoning in me.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

Crowd Analysis is fundamental to solving many real-word problems. It is important for management of crowded events, such as protests, demonstrations, marathons, rallies, political speeches and music concerts which are characterized by gatherings of hundreds to thousands of people. It has use in the design of public spaces and infrastructure, as well as in their expansion and modification, by analyzing the counts of customers and commuters that frequent and travel through these places. It has applications in computer graphics as well, where crowd simulation models can be learned using data from real-world crowded scenes. But, perhaps its most important use is in visual surveillance and anomaly detection. The recurrent and tragic stampedes at pilgrimages [1] and parades [2] as well as the recent terrorist attack at a marathon [3] call for improved and sophisticated techniques for visual analysis of dense crowds. These include human detection and tracking, counting, action recognition, anomaly detection and classification of high-level events.

A crowd is more than the sum of individuals; the difficulty of computer vision tasks increases disproportionately depending on the number of individuals making up the crowd. This can be gauged by the fact that the human response to an image of a crowd is much slower than that on a non-crowd image. For instance, a human or a member of surveillance team can easily detect, track and count in an image of a few people, but when presented with a crowd image containing hundreds to thousands of people, will require a considerably large amount of time. Thus, the straightforward extension of computer vision algorithms does not yield corresponding improvement [4, 5]. Furthermore, the applicability of a particular computer vision algorithm also depends on the density of the crowded scene [6]. Scenes of high density crowds can be divided into groups based on number of pixels on target, and those with extremely small object size permit only holistic approaches for scene understanding, such as finding motion patterns and segmentation of crowd flows [7, 8, 9, 10] as well as counting and localization. However, if individuals in a crowd are distinguishable then detection and tracking of individuals may be possible, which is important in the context of safety and surveillance [6].

Dense crowds offer a set of challenges when it comes to visual analysis - fewer pixels per target, perspective effects and severe occlusions. But, they also provide constraints which can be employed to tackle these challenges. These can be both contextual (spatial) or temporal constraints. For instance, tracking methods for dense crowds learn the crowd flow, and use that flow to reliably track individuals in the crowd [11, 12, 13]. Such use of repetition of behavior in time is largely exclusive to dense crowds. In this dissertation, we pursue an alternative direction and explore the use of spatial or contextual constraints in dense crowds to improve counting, detection and tracking.

1.1 Counting and Localization in Images of Dense Crowds

The manual counting of individuals in very dense crowds is an extremely laborious task, but is performed nonetheless by experienced personnel when needed [14]. Computer vision research in the area of crowd analysis has resulted in several automated and semi-automated solutions for density estimation and counting. Practical application of most existing techniques however, is constrained by two important limitations: (1) inability to handle crowds of *hundreds or thousands* (Fig. 1.3) rather than a few tens of individuals [15, 16]; and (2) reliance on temporal constraints in crowd videos [17], which are not applicable to the more prevalent *still images*.

The proposed approach for counting is motivated by the fact that in extremely dense crowds of people, no single feature or detection method is reliable enough to provide an accurate count due to low resolution, severe occlusion, foreshortening, and perspective. Indeed even the state-of-the-art human, head, or face detectors perform poorly in such scenarios. We observe however that densely packed crowds of individuals can be treated as a texture, albeit irregular and inhomogeneous at a coarse scale. And this texture begins to correspond to a harmonic pattern, as is the case in regular textures, at a finer scale. Furthermore, there does exist a spatial relationship that is expected to constrain the counting estimates in neighboring local image regions in terms of similarity of counts.

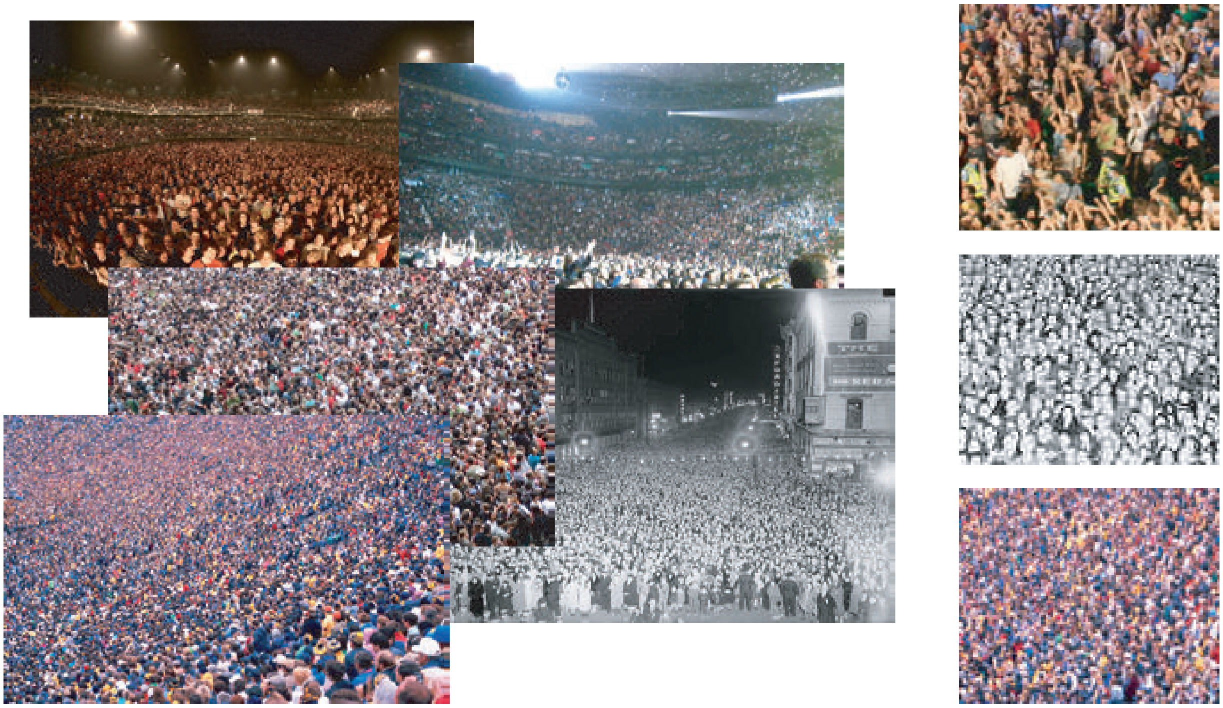


Figure 1.1: This figure shows five arbitrary images from the dataset used for crowd counting and localization. On average, each image in the crowd counting dataset contains around 1280 humans. The column on the right shows four patches from different images at original resolution.

We also observe that, in derived intensity spaces such as image derivatives or edges, groups of individuals are likely to exhibit an increased level of similarity. Therefore, in addition to supervised training of human or head detectors, appearance based feature descriptors like SIFT are also useful to estimate the so called texture elements or textons [18]. This observation has been used successfully for crowd detection in [19], although not for counting or localization. Our goal in using appearance based descriptors for localized patches is to estimate repeating structures in the image, but with the important distinction that such image patches are not expected to fully contain a person, rather the textons can represent a single part of a person, multiple parts, or multiple people and their parts.

Another main contribution of the proposed framework for counting is the use of frequency-domain analysis in crowd counting. Fourier transform has been used extensively in texture analysis [20], and specifically in crowd analysis [21]. Given geometrically arranged texture elements, the Fourier transform can provide reliable estimates of the texton counts [22]. In the domain of crowd counting however, the application of frequency analysis is severely limited due to two main reasons: (1) the spatial arrangement of texture elements is very irregular; and (2) the Fourier transform is not useful in localizing the repeating elements.

We propose novel solutions to overcome these limitations. First, we employ Fourier analysis along with head detections and interest-point based counts in local neighborhoods on multiple scales to avoid the problem of irregularity in the perceived textures emanating from images of dense crowds. The count estimates from this localized multi-scale analysis are then aggregated subject to global consistency constraints. Secondly, in order to leverage multiple estimates from distinct sources, the corresponding confidence maps need to be comparable and in the same space. For instance, the Fourier transform is not directly useful in this regard since it cannot be combined with count estimate maps in the image domain. We therefore reconstruct the low to medium frequency component of image region and the reconstructed image is then compared with the original image after alignment. This process provides two important pieces of information: the estimated count per local region, and a measure of error relative to the original image.

Combining the three sources, i.e., Fourier, interest points and head Detection, with their respective confidences, we compute counts at localized patches independently, which are then globally constrained to get an estimate of count for the entire image. Since the data terms are evaluated independently at different scales, the smoothness constraint has to be applicable to spatial neighborhoods as well as immediate neighbors at different scales. We propose a solution to obtain counts from multi-scale grid MRF which infers the solution simultaneously at all scales while enforcing the count consistency constraint.

The proposed approach for counting is only capable of estimating the counts in an entire image. Several tasks aimed at automated analysis of dense crowds require the localization of humans where the goal is to pinpoint the position of all humans in the image. In an image of extremely dense crowd, usually the heads are visible while the bodies are occluded, the problem then reduces to locating the head of each individual in the image. Therefore, we also present an approach that can localize the heads of humans in still images. Figure 1.2 gives an illustration of the problem.

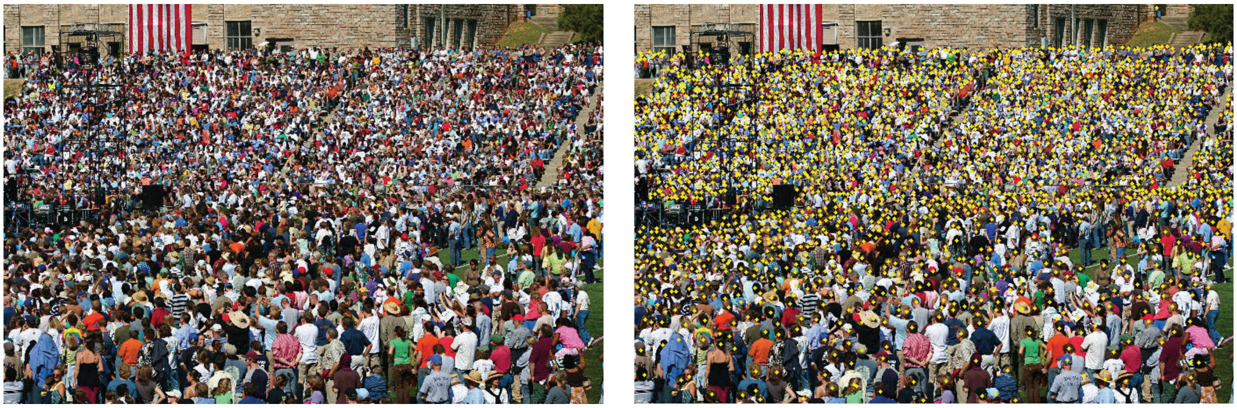


Figure 1.2: This figure illustrates the idea of localization. Given an image such as the one on the left, our goal is to locate the position of each and every person in the image as shown with yellow dots on the right.

Localization has an additional advantage for counting as well. For a real user or analyst who wants to estimate the exact count for a real image *without any error*, the results of counting alone are not sufficient for this task. The single number for an entire image makes it difficult to assess the error or the source of the error. However, if the user is supplied with dotted locations of the individuals, then it is possible to quickly go through the image and remove the false positives and add the false negatives. The count using such an approach will be much more accurate and the user can get 100% precise count for the query image.

To address this difficult problem, we propose to utilize the inherent structure of a crowded scene. One of the characteristic features of the crowd is repetition of patches corresponding to visible parts of humans, typically the heads. As we will see in Chapter 3, this repetition is far from periodic. In fact, it is actually described by a random process. Thus, it is not possible to discover any pattern using regularity in spatial locations. Nonetheless, the building block or the atom of such a pattern can be detected with some reliability through head detectors. The task then transforms from discovery of the pattern to its expansion with initialization provided by head detections. We propose several criteria to select such atoms. Once the hypotheses are expanded using correlation, they are then selected using binary integer quadratic programming subject to Special Ordered Set (SOS) Type 1 constraints. Every hypotheses induces one such constraint and limits the selection of other competing hypotheses in its neighborhood. The experiments are performed on the same dataset used by our approach for counting since the ground truth for each image in the dataset has dotted annotations defining the exact location of each individual.

1.2 Detecting humans in Dense Crowds using Locally-Consistent Scale Prior and Global Occlusion Reasoning

The methods introduced in previous section for counting and localization are applicable to images of extremely dense crowds containing thousands of people. High-resolution images of medium to high density crowds containing tens to hundreds of people permit human detection in terms of bounding boxes that cover the extent of each human in the image. Although there has been considerable research on detecting humans in images containing a few individuals [23, 24, 25, 26, 27], the existing methods do not perform well on crowded scenes primarily because of severe occlusions, where significant parts of humans are invisible in the image due to occlusions from other humans. The other issue is clutter due to the presence of significant edge information (image gradients) spread throughout the image. This introduces noise for the human detector, thus increasing the number of false alarms. To tackle this problem, we propose to weigh detection hypotheses using scene-derived priors that are discovered automatically from a given image.

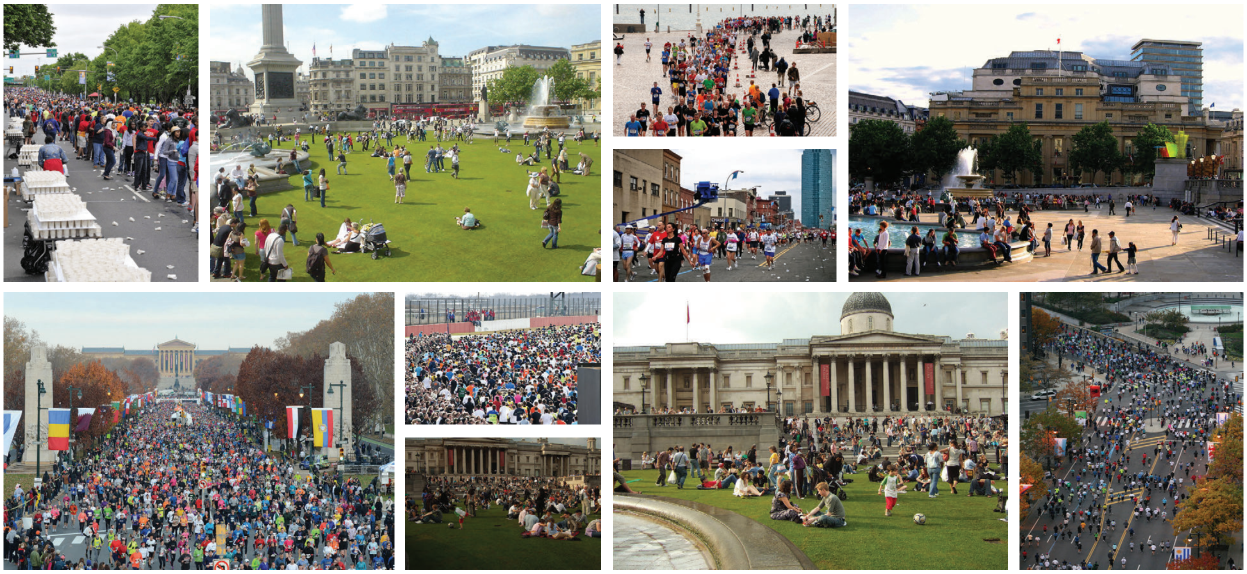


Figure 1.3: This figure shows several images from the dataset on which experiments for human detection were performed. Although there are occlusions and pose variations, the consistency in scale is evident in all images which can be used to restrict the space of detection hypotheses in these images.

Considering the images in Fig. 1.3 which depict crowds of different densities, it can be observed that the scale or size of neighboring individuals is similar to each other. Although the scale changes across all the images, the change in scale is gradual due to the perspective effect and overhead position of camera. Even when the camera is not located overhead, the scale is still locally consistent but with sharp discontinuities due to humans at various depths. In our approach, we propose to embed these qualitative observations in a discontinuity-preserving Markov Random Field that captures the scale and confidence of humans in an entire image. For each pixel in the image, the scale prior captures the size of human that is expected to occur at that location. Similarly, the confidence prior models the probability of occurrence of a human in that particular part of the image.

Similarly, there can be several heuristic based methods for occlusion handling, but such methods are not applicable in dense crowds, as there never are isolated pairs of individuals which have to be reasoned for occlusion - individual A may occlude B and individual B may occlude C and so on, making all of them tied to each other. An incorrect solution for one individual in a greedy algorithm can affect detection of many other individuals. Thus, instead of developing a greedy or heuristic solution, we leverage the advanced solutions of Integer Programming to simultaneously reason all occlusions and infer visible areas of detections. Unlike, non-crowd human detection methods that detect and localize humans in isolation, our approach solves the problem in a global fashion, thereby, honoring the relationship that individuals in a dense crowd have with each other.

The key ideas presented for proposed human detection are independent of underlying human detector used, but due to its popularity, we used Deformable Parts Model [24] to obtain the scales and confidences of humans and their component parts, which are then used by the proposed method. Since individuals in dense crowds undergo severe occlusions, and full-body human detection is not sufficient to detect all humans, part-based analysis consequently assumes a greater significance in such scenes. We propose a solution to detect combinations-of-parts of humans which is able to increase recall by detecting partially occluded humans - a common occurrence in dense crowds. This allows us to have multiple detectors that use same parts, which spares us from part-specific annotations and is computationally efficient as it reuses results of filter responses on the shared parts.

Our approach bridges the gap between holistic approaches to crowds and isolated analysis of individuals in non-crowded scenes. The contributions of this approach can be summarized as follows: 1) use of locally-consistent scale prior for human detection and an approach for its application in dense crowds, 2) a method to create detectors comprising multiple parts without requiring annotations of those parts, made possible through the use of latent SVM, 3) occlusion reasoning in crowds with a global solution, and 4) an evaluation on a new and challenging dataset of dense crowd images with tens of thousands of annotated humans.

1.3 Tracking using Prominence and Neighborhood Motion Concurrence

Tracking in dense crowds [11, 28, 29] is a challenging problem because the large number of objects in close proximity pose difficulty in establishing correspondences across frames. Methods specifically designed for dense crowds generally require some learning of motion priors, which are later employed for tracking. For instance, the first popular method for the problem [11] was based on the assumption that all individuals in a crowd behave in a manner consistent with global crowd behavior. The authors learn the direction of motion at each location in the scene, termed floor fields, and use that to predict motion of individuals while tracking. The floor fields severely restricts the permitted motion that individuals in a particular scene can have. This restriction on the motion of individuals due to time-invariant priors [11, 30] would cause the tracker to fail when, (1) the crowd flow is dynamic, (2) the crowd flow shifts or moves to a new region which was not learned before, and (3) when there are anomalies. Furthermore, camera motion and jitter can make learning the crowd flow difficult, if not impossible. Though learning, whether online or offline, certainly helps in tracking dense crowds when these issues are not present, we emphasize the use of visual and contextual information available in such crowded scenes to track in an online manner, without any pre-processing, learning or crowd flow modeling.

At the core of our approach lies template-based tracking, which is used to obtain probability of observation. However, the simplicity of a template-based tracker demands more than just appearance to perform well in high density crowds. We supplement the tracker with novel visual and contextual sources of information, which are particularly relevant to crowds and reduce the confusion in establishing correspondences.

The first idea we explore is prominence of individuals which is similar to saliency (generally used for features and points). In any crowded scene with a large number of people, the appearance of some individuals will be markedly different from the rest (Figure 1.4). The probability of confusing such individuals with the rest of the crowd will be low. Thus, the prominence of such individuals provides extra information which we propose to leverage in tracking.

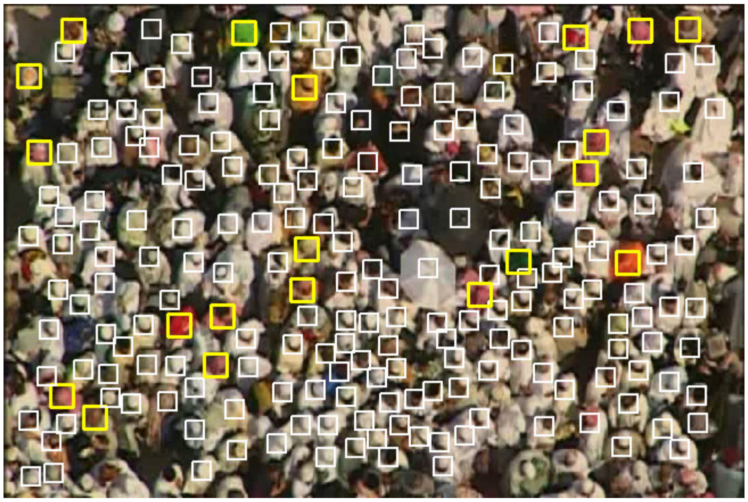


Figure 1.4: An example of a dense crowd where individuals that are in yellow squares stand out from the crowd and, therefore, should be easier to track than rest of the individuals in white squares.

The second idea is to employ influence from neighbors to make better prediction of an individual's position. This idea is based on the observation that individuals in dense crowd experience social forces that bound their movement [31]. For instance, an individual cannot jump across his neighbors in a single time instance. The restriction on movement that each individual experiences is proportional to the density of the crowd. Social force models, both in computer graphics and vision, are generally geared towards collision avoidance where the goal is to predict positions such that subjects or individuals avoid collisions with each other. Our model, on the other hand, exploits the fact that movement of individuals in a dense crowd is similar to their neighbors, and therefore can be used to make better predictions.

Combining prominence and influence from neighbors, our method imposes an order on the way positions of individuals are updated. Individuals with prominent appearance are updated first, who subsequently guide the motion of the rest of the crowd in a manner similar to constraint propagation. While updating, if the underlying patch-based tracker gives weak measurement for an individual, then the position of that individual is updated based on appearance-based dense instantaneous flow. Thus, the framework we introduce incorporates these ideas, as well as their inter-relationships. Our contributions for this approach can be summarized as,

- An alternative approach to dense crowd tracking which highlights the significance of prominence and spatial context for tracking dense crowds without requiring crowd flow modeling,
- Introduction of the notion of prominent individuals, its relevance to tracking in dense crowds, and a method to detect prominent individuals,
- Incorporation of influence from neighbors, prominent or not, to better predict and estimate an individual's position,
- A tracking framework which imposes an order on the way individuals are tracked, where positions of prominent individuals are updated first and individuals with low probability of observation from underlying tracker are updated last.

Since space is complementary to time, both the visual information (prominence) and spatial context (influence from neighbors) are complementary to temporal constraints (crowd flow, motion patterns) introduced in previous works on tracking people in dense crowds. Our goal in this work is to emphasize the first two, which when coupled together allow tracking in an online fashion, without requiring crowd flow modeling or observations from the future.

1.4 Dissertation Organization

The rest of the dissertation is structured as follows: In Chapter 2, we review existing literature on automated visual analysis of dense crowds. In Chapter 3, we present our approaches on two related problems of counting and localizing humans in images of extremely dense crowds. Chapter 4 provides our approach for human detection using scene-derived priors and global occlusion reasoning on crowded images of medium to high density. In Chapter 5, we present our method to detect salient individuals in a crowd video which in addition to Neighborhood Motion Concurrence allows us to track individuals simultaneously. Finally, Chapter 6 provides the conclusion and a discussion on the identified directions for future works.

CHAPTER 2: LITERATURE REVIEW

Crowd analysis is an active area of research in Computer Vision [6]. Over the past few years, methods have been proposed that estimate density and number of people in a crowd [32, 33], find group structures within a crowd [34], detect abnormalities [35, 36, 37, 38], find flow segments [9, 10], and track individuals in a crowd [39, 28, 29]. Some methods proposed in literature for crowd detection perform image segmentation without actual counting or localization [19], while others simply estimate the coarse density range within local regions [40]. Here, we focus on methods that solve the three main problems addressed in this dissertation: counting, detection and tracking.

2.1 Counting humans in Surveillance Imagery

Most of the existing algorithms for estimating exact counts of humans have been tested on low to medium density crowds, e.g., USCD dataset with density of $11 - 46$ people per frame [15], Mall dataset with density of $13 - 53$ individuals per frame [16], and PETS dataset containing $3 - 40$ people per frame [41]. In contrast to these images and videos, we focus on still images containing between 94 and 4543 humans, with an average of 1280 individuals per image. Such high density implies that an individual may occupy so few pixels that it can neither be detected nor can its presence be verified given the location, which are key requirements in existing techniques.

Person detection for counting individuals, present in an image or video, has been employed in [42, 43]. This category of methods however is not useful for the kind of images we deal with, because human, or even head and face detection in these images is difficult due to severe occlusion and clutter, low resolution, and few pixels per individuals due to foreshortening. Brostow and Cipolla [44] and Rabaud and Belongie [32] count moving objects by estimating contiguous regions of coherent motion. Computation of such patterns of motion were also proposed in [45, 46, 47], but not with explicit application to the problem of crowd counting. These algorithms require video frames as input, with reasonably high frame rate for reliable motion estimation, but are not suitable to still images of crowds, or even videos if the individuals in the crowd show nominal or no motion, e.g., political gatherings and concerts.

Another category of techniques proposed for crowd counting rely on estimation of direct relationships between low level or local features and counts by learning regression functions. Such a function can be global [15, 48, 49, 50] where a single function's parameters are learned for the entire image or video. These methods have the implicit assumption that the density is roughly uniform regardless of the location where the feature is computed. This assumption is largely invalid in most real world scenarios due to perspective, changes in viewpoint, and changes in crowd density.

The problems associated with global feature regression can be alleviated by relaxing this assumption. Methods such as [51] propose to divide an image into cells and perform regression individually for each cell. These methods [51, 52] aim to compensate for problems associated with foreshortening, and local geometric distortions due to perspective. One key problem with this approach however is that the local context, or spatial consistency constraints are ignored as information across local regions is not shared.

Chen et al [16] recently proposed that information sharing among regions should allow more accurate and robust crowd counting. They propose a single multi-output model for joint localized crowd counting based on ridge regression. Their proposed framework employs interdependent local features from local spatial regions as input and people count from individual regions as multi-dimensional structured output. The algorithm however was not applied to scenarios with crowds of more than a few tens of people.

2.2 Human Detection

Human detection is often the precursor to many computer vision tasks and the problem has been tackled by various approaches in literature [23, 24, 25, 26, 27, 53, 54, 55, 56, 57, 58]. A recent comprehensive survey by Dollar et al. [59] compares various state-of-the-art pedestrian detectors and evaluates their performance based on scale, degree of occlusion and localization accuracy. They conclude that under partial occlusion the performance degrades significantly, and becomes *disappointing* at low resolutions and under partial occlusions. The authors make an assessment that there is still a considerable gap between the current and desired performance of these human detectors. However, they do suggest that the use of some form of context and better occlusion handling can improve the performance of detectors. Another survey from the perspective of traffic safety is by Geronimo et al. [60] which focuses on application of pedestrian detection to assist drivers, with the goal of avoiding possible accidents and casualties.

Human detection poses a range of challenges, the most important of which are to deal with articulation and handle occlusions. The non-rigid structure and deformity in humans is modeled using the notion of constituent parts which allow certain degree of displacement of parts relative to their desired positions. Several part-based approaches have been proposed in the literature [24, 61, 62, 63, 64]. In [24], the part filters are learned and applied individually, with each part placed relative to the root location and a deformation cost added to the final confidence. Similarly, some of the recent approaches have used the visibility of parts to infer the occluded regions. Ouyang and Wang [65] model the visibility of parts as hidden variables in a probabilistic framework, whereas, Enzweiler et al. [66] use mixture-of-expert classifiers and train them on features from intensity, depth, and motion to handle partial occlusions. Duan et al. [67] describe the relations between parts using manually defined rules in a hierarchical structure of words, sentences and paragraphs to deal with articulation and occlusions. Wang et al. [26] train a HOG-LBP/SVM classifier and present a method to find contributions from individual parts, which are used to construct an occlusion map depicting visible regions in detections. We use a similar approach for dividing the bias term among parts, but unlike rigid parts used in [26], we have to cater for deformation costs in the DPM framework.

State-of-the-art human detectors perform reasonably well to handle deformation and mild occlusions in non-crowded scenes. However in dense crowds, where individuals undergo severe occlusions, large deformations as well extreme variations in apparent size, human detection becomes an extremely challenging task. For detection in low-density crowded scenes, a unified probabilistic framework by Yan et al. [68] uses appearance and spatial interaction to describe multiple pedestrians. Improved occlusion handling using a multi-view geometry approach is presented by Ge and Collins [69], who estimate the number of people in a crowd and their locations by sampling from a posterior distribution over a 3d crowd configuration. Crowd density is utilized by Rodriguez et al. [17] who show improved person localization and tracking performance in crowded scenes. They formulate the problem as an optimization of a joint energy function by incorporating confidences of detections subject to overlap and scene-specific density constraints. A video is divided into two sets, where the first set with annotated humans is used to train the density estimator, while the second set is used for testing. The ideas presented for our approach are complementary to [17], but our goal is to use cues or priors that are generally applicable, and not learned from and applied on, individual scenes or videos.

Human detection is a pre-requisite to tracking, but due to the difficulty in detecting humans in dense crowds, approaches rely on temporal repetition in the form of motion patterns [70, 71] and floor fields [11] to track and analyze crowded scenes, and require manual initialization of tracks [71, 70]. Ali and Shah [11] use this idea in the form of static, dynamic and boundary floor fields, which determine the probability of motion from one location to another. Crowd behavior has been similarly modeled by Rodriguez et al. [12] in unstructured scenes to track individuals. Rodriguez et al. [72] use a large collection of crowd videos to learn motion patterns which are then used to drive a tracking algorithm. Multi-target tracking combined with motion pattern learning by Zhao et al. [71] has shown to improve tracking in structured crowds. It requires user labeling of the target in the first frame, which is used to learn a detector, later employed to detect and track other similar objects in the sequence. The common theme in these works is temporal modeling of crowd motion and manual initialization of individual tracks. Thus, improved human detection can help reduce manual initialization related to tracking dense crowds.

Inspired from human visual system which makes use of contextual information to detect and recognize objects, context in computer vision has been extensively studied and used to improve object detection. Researchers have experimented with various approaches: semantics [73], image statistics [74], shape context [75], pixel context [25, 76] and color/texture cues [77], 3D geometric context [78] as well as intensity/depth/motion cues [66]. Divvala et al. [79] evaluate several sources of context and propose the use of geographic context and object spatial support. The work by Desai et al. [80] focuses on learning spatial context to simultaneously predict labeling of a scene while bypassing heuristic-based post-processing steps. Similarly, Ding and Xiao [81] combine the local window with neighborhood windows to construct a multi-scale image context descriptor from HOG-LBP features. This shows that the information required to detect an object not only resides in the extracted features, but also within the scene to which the object belongs. For our approach, we propose to use context in the form of locally-consistent scale prior which enforces the constraint that the size of proximal individuals in a dense crowd is consistent and similar, though there may be occasional discontinuities. Closely related with scale prior is the confidence prior which gives the confidence of associated scale at each location in the image. We show that both these priors can be automatically discovered from the scene and are extremely relevant to detecting humans in dense crowds.

2.3 Tracking of humans in Videos

Methods proposed for multi-target tracking include Park et al. [82] who sped up belief propagation using mean shift by sparsely sampling the belief surface instead of using parametric methods or non-parametric methods that require dense sampling. They do not assume prominence and pass messages in all directions, therefore presuming absence of anomalies. There are a few papers that have used context for tracking as well. Yang et al. [83, 84] used contextual information to improve the tracking performance of a few objects. Through color segmentation of the image, they find auxiliary objects, which are easier to track and whose motion is correlated with the target. The auxiliary objects are then tracked; they also aid in tracking the target, which occurs simultaneously. The method was streamlined for non-crowd scenarios, with results containing a maximum of three objects per sequence. Furthermore, due to hundreds of people frequently occupying the entire screen in crowd videos, the definition and discovery of auxiliary objects is not applicable in crowd sequences. Khan et al. [85] also capture interaction between targets using particle filters in an MRF framework. However, they do not consider prominence and anomalies while tracking, and the particle filters are not suitable for crowd sequences due to fewer pixels per target.

Next, we review papers that use salient object and social force model for tracking followed by extensive review of methods designed for tracking in dense crowds. For an in-depth analysis, interested readers are referred to the survey by Zhan et al. [6].

Prominence. Discriminative features were used for tracking by Collins and Liu [86], who rank the foreground features online, and track objects using only those features which discriminate foreground from background. A similar idea was explored by Mahadevan and Vasconcelos [87] who, given a pool of features from foreground and background, select the most informative features for classification between the two. In relation to our method, prominence can be seen as a *collection* of salient features which discriminate one foreground object from the rest.

Social Force Models. Static motion models (such as linear velocity or constant acceleration) have long been used for tracking in computer vision. Dynamic models, as opposed to static ones, account for the dynamic structure of the scene and objects, and are based on the fact that individuals are driven by goals and respond to changes in their environments by adjusting their paths. Methods that model [88, 89, 90] and simulate crowds [91] incorporate this crucial information to produce realistic results.

In computer vision, social force models have been used for multi-target tracking, such as Pellegrini et al. [92]. They introduced Linear Trajectory Avoidance, a model inspired by Helbing and Molnár [31], in which predictions are made so that individuals avoid collisions with each other and the obstacles. The repulsive forces are balanced by a preference of each individual to move towards a destination with some desired speed, both of which are assumed to be known in advance. The experiments were performed on non-crowded scenes, since collision avoidance has lower applicability to dense crowds where individuals have less freedom of movement. To overcome some of the shortcomings in [92], Yamaguchi et al. [93] proposed a similar approach using a more sophisticated model, which tries to predict destinations and groups among individuals, using certain heuristics based on trajectory features and a classifiers trained on annotated sequences. They tested on very simple scenes, and assume people move along straight paths as the destination cannot change with time. There are scenes where this assumption will break, for instance, Sequence 5 in Section 5.5.

Furthermore, Yamaguchi et al. [93] penalizes deviations from preferred speed, which is set to 1.3 m/s. This is the speed at which an average human walks, but this constant will be different for a scene depicting a marathon, where people can be seen running at various speeds. In fact, both the methods [92, 93] assume that the positions of individuals are in metric space, where distances can be computed between individuals in terms of metric units (meters). This is a natural disadvantage of sophisticated social force models whose parameters, otherwise, would have to be learned for each testing video anew. Secondly, for correct transformation of positions of individuals from image space to metric space, both methods assume that the video be captured from bird's eye view even if there are only a few individuals at any given time. These two strict assumptions limit their applicability to arbitrary videos. Although some camera elevation is necessary to completely capture a dense crowd, the proposed model which works in image space can work with slightly slanted views, i.e, lower than bird's eye, because we anchor motion of all individuals on prominent ones who lie in the same scene as rest of the crowd. In other words, since we impose motion consistency in image space, we do not require to know the transformation between image and metric coordinates, as such transformations cannot be assumed to be known in advance for arbitrary videos.

Dense crowds. Recently, Garg et al. [94] addressed the problem of matching instances of people in images of crowded events using photographs from Flickr. Unlike our problem, which deals with single-view videos, their method works on images taken from the same scene which allows structure from motion and 3d reasoning to match subjects.

For tracking in dense crowds, in a series of papers, Kratz and Nishino [70, 95, 13] trained Hidden Markov Models to learn motion patterns in the scene which they later use for tracking individuals. Our method provides an alternative to such training-based methods by using appearance and contextual information only. The method proposed by Song et al. [29] tracks individuals by learning patterns of flow through online clustering of tracked trajectories. Wu et al. [28] did not learn any priors but employed multiple cameras to obtain 3D trajectories of objects that are indistinguishable in terms of appearance by finding correspondences across the multiple views.

The works most similar to ours is that of Ali and Shah [11], who use transition probabilities computed from learned floor fields, in order to track individuals in a dense crowd. The method requires a pre-processing period where the static floor field is learnt using particles advected through optical flow across the scene. Furthermore, the dynamic floor field which captures the instantaneous flow is a non-causal process, as it uses observations from the future. Similarly, Rodriguez et al. [30] use Correlated Topic Model (CTM) to capture different overlapping and nonoverlapping crowd behaviors in the scene. In their construction, words correspond to low-level quantized motion features and topics correspond to crowd behaviors. Similar to [11], the method requires temporal modeling of crowd behavior which uses observations from the future. Recently, Rodriguez et al. [72] proposed a method that solves the same problem, but instead of learning crowd flow, they build a database of approximately five hundred videos and match patches from query videos to the database videos. Their method requires extensive searching of similar patches in database, while making a strong assumption that the motion of individuals in a particular query patch can be found in database. We, on the other hand, rely completely on information that is readily available in the sequences.

Different from previous approaches, our goal is to develop an online tracker for dense crowds without requiring extensive analysis of sequences in the database, or off-line analysis by modeling the crowd behavior in advance. Instead, we explore visual and spatial information in this work in the form of prominence and influence from neighbors while making sure that the method is not biased against anomalies or dynamic crowd flow like the previous methods. Since temporal information is complementary to spatial and visual constraints, the proposed method can be seen as an alternative and complementary approach to previous methods for tracking individuals in videos of structured dense crowds.

Furthermore, due to difficulty of human detection in dense crowds, and to keep the primary focus on tracking, all previous works in this area [11, 30, 72, 13] assume that a manual initialization of templates on individuals in the crowd is afforded to the algorithm. The template refers to a bounding box around the individual we intend to track. For our approach, like previous works, we also assume that initial templates (bounding boxes) are provided and our goal is to track them across the scene. This also restricts the applicability of other social force [96] or tracking methods which perform data-association among human detections across frames of the video.

CHAPTER 3: COUNTING AND LOCALIZATION IN DENSE CROWDS

This chapter focuses on the two related problems of counting and localization in images of extremely dense crowds. Given an image, our goal for counting is to estimate the number of people in the image, whereas for localization, the problem is to pinpoint the location of each person. The density of people, i.e., the number of people per unit area, in an arbitrary crowded image is rarely uniform, and varies from region to region. This variation in density may be inherent to the scene that the image captures (different distribution of individuals in different parts of the scene) or it may arise due to the viewpoint and perspective effects of the camera. Due to these reasons, an extremely dense crowded scene cannot be analyzed in its entirety for either counting or localization. Therefore, our approach to both problems involves dividing the scene or image into small patches, performing the analysis locally, and merging the results afterwards.

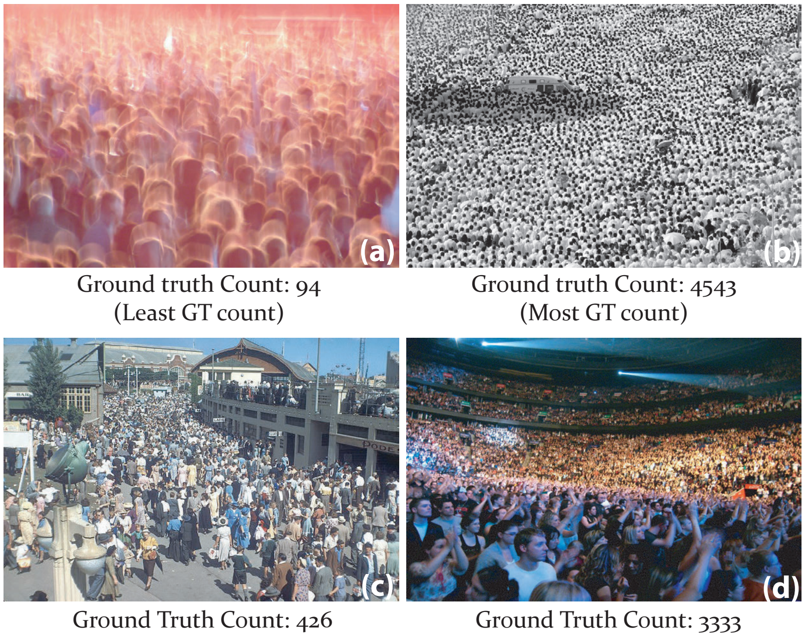


Figure 3.1: This figure shows four images selected from the dataset. The images in (a) and (b) have the lowest and highest ground truth count respectively.

For evaluating our approaches to both problems, we collected a new dataset from publicly available web images, including Flickr. As mentioned in the introduction, it consists of 50 images with counts ranging between 94 and 4543 with an average of 1280 individuals per image. Much like the range of counts, the scenes in these images also belong to a diverse set of events: concerts, protests, stadiums, marathons, and pilgrimages. One of the images is a painting while another is an abstract depiction of a crowd (the one with the least count, shown in Fig. 3.1a). Using a simple tool for marking the ground truth positions of individuals, we obtained 63705 annotations in the fifty images. Some examples of images with the associated ground truth counts can be seen in Fig. 3.1.

3.1 Multi-source Multi-scale Counting in Crowded Images

For counting, the proposed framework begins by estimating the number of individuals in small patches uniformly sampled over the image to cater for the change in density. Even though the density of the crowd, i.e., the number of people / pixels^2, varies across the image, it does so smoothly, suggesting the density in adjacent patches should be similar. We handle the issues of variation in density and smooth variation separately. When counting people in patches, we assume the density is uniform but implicitly assume that the number of people in each patch is independent of adjacent patches. Once we estimate density or counts in each patch, we remove the independence assumption and place them in multi-scale Markov Random Field to model the dependence in counts among nearby patches.

3.1.1 Counting in Patches

Given a patch P, we estimate the counts from three different and complementary sources, alongside confidences for those counts. The three sources are later combined to obtain a single estimate of count for that patch using the individual counts and confidences.

3.1.1.1 HOG based Head Detections

The simplest approach to estimate counts is through human detections. However, a quick glance at images of dense crowds reveals that the bodies are almost entirely occluded, leaving only heads for counting and analysis. We, therefore, used Deformable Parts Model [23] trained on INRIA Person dataset, and applied only the filter corresponding to head to the images. Often, the heads are partially occluded, so we used a much lower threshold for detection. There are many false negatives and positives since the images are inherently difficult (see Fig. 3.2). The detections are accompanied with scale and confidence. For each patch, we use number of detections, η_H, mean and variance of scale $\mu_{H,s}$, $\sigma_{H,s}$ and confidence $\mu_{H,c}$, $\sigma_{H,c}$. The consistency in scale and confidence is a measure of how reliable head detections are in that patch.

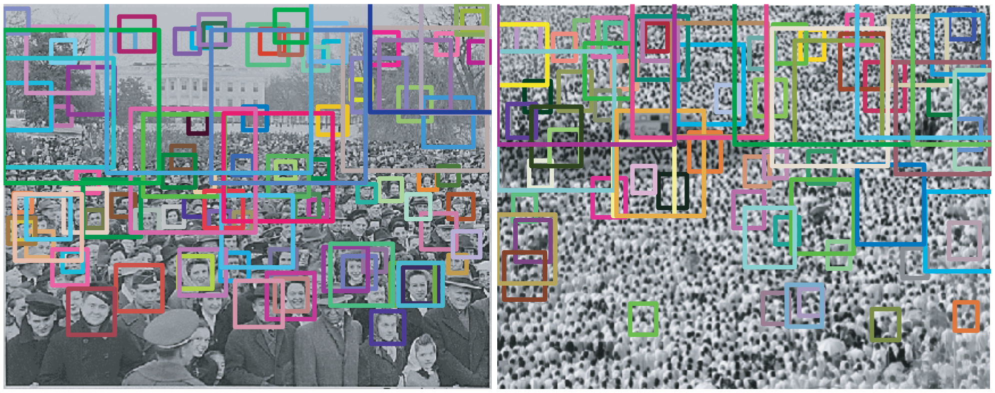


Figure 3.2: Results of Head Detection: Image on the left is one of the few images where head detection gives reasonable results. False negatives and positives are still evident in both images.

3.1.1.2 Fourier Analysis

When a crowd image contains thousands of individuals, with each individual occupying only tens of pixels, especially those far away from the camera in an image with perspective distortion, histograms of gradients do not impart any useful information. However, a crowd is inherently repetitive in nature, since all humans appear the same from a distance. The repetitions, as long as they occur consistently in space, i.e., crowd density in the patch is uniform, can be captured by Fourier Transform, $f(\xi)$, where the periodic occurrence of heads shows as peaks in the frequency domain. Specifically, for a given patch, we compute the gradient image, $\nabla(P)$, and apply a low-pass filter, $f(\xi) = 0, \forall \xi > \xi_o$, to remove very high frequency content. Next we discard low amplitude frequencies, which is followed by reconstruction, P_r, through inverse Fourier Transform. We find the number of local maximas in the reconstructed image (Fig. 3.3) after alignment and non-maximal suppression which serves as an estimate for the Fourier-based count, η_F. In addition, we compute several other measures, such as entropy as well as statistical measures related to first four moments - mean, variance, skewness and kurtosis for both the reconstructed image and difference image $|P_r - \nabla(P)|$. The count is normalized for the size of the patch.

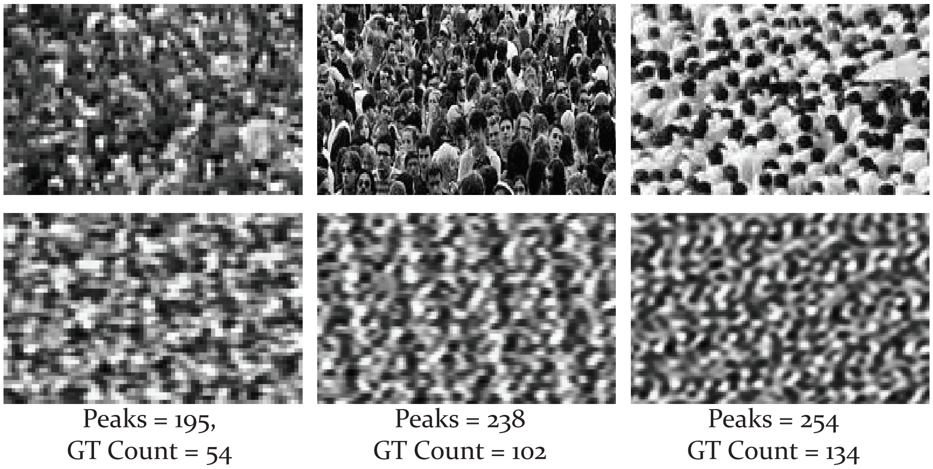


Figure 3.3: Counting through Fourier Analysis: The first row shows three original patches, while the second row shows corresponding reconstructed patches. The positive correlation is evident from the number of local maximas in the reconstructed patch, and the ground truth counts shown at the bottom.

3.1.1.3 Interest Points based Counting

We use interest points not only to estimate counts but also to get a confidence whether the patch represents crowd or not. Since sky, buildings and trees naturally occur in outdoor images, and the fact that head detection gives false positives in such regions (Fig. 3.2) and Fourier Analysis is crowd-blind, it is important to discard counts from such patches. For both counting and confidence, we obtain SIFT features, and cluster them into a codebook of size c. In order to obtain counts or densities using sparse SIFT features, we use Support Vector Regression using the counts computed at each patch from ground truth.

From the perspective of Statistics, the number of individuals in a particular patch can be seen as spatial Poisson Counting Process with parameter (corresponds to density), λ, i.e., $N(P) \sim \texttt{Poisson}(\lambda|P|)$, where $|P|$ gives the area of the patch P in pixels. The expected value of $N(P)$ is simply $\lambda|P|$. Since we assumed the density is uniform in the patch, the process is homogenous and λ is not a function of location (x, y). Moreover, the independence assumption among patches gives, for the image, I:

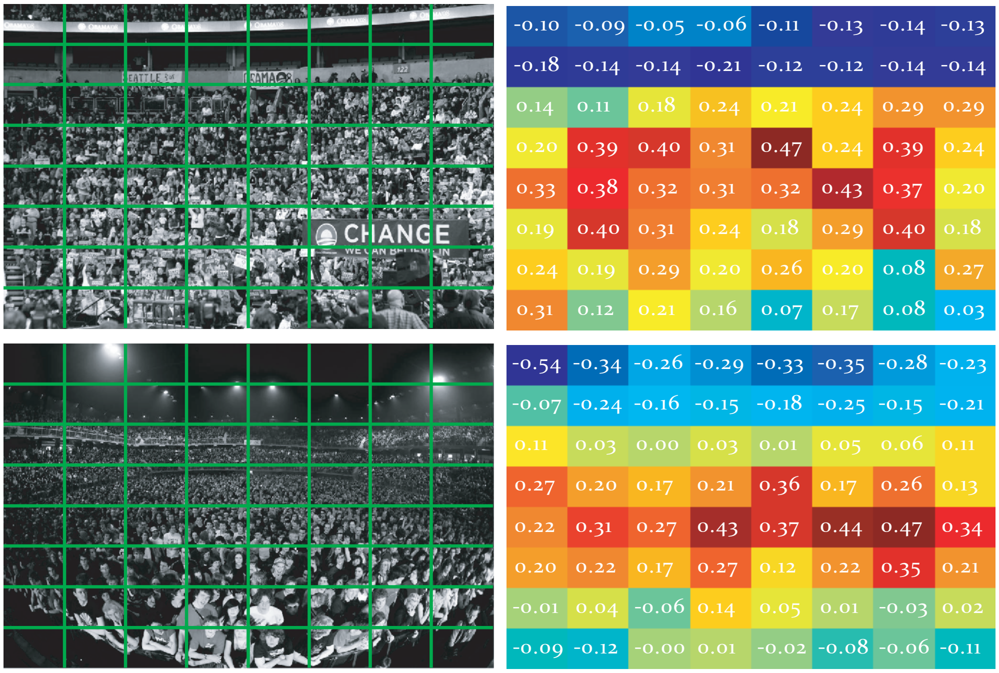
$$\begin{aligned} N(I) &= N(P_1 \;\cup\; P_2 \;\ldots\; P_n) \\ &= N(P_1) + N(P_2) + \ldots + N(P_n), \end{aligned} \tag{3.1}$$

where $P_1, P_2, \ldots P_n$ form a disjoint partition of I.

Furthermore, due to sparse nature of SIFT features, the frequency γ of a particular feature i in a patch can also be modeled as a Poisson R.V., $p(\gamma_i|crowd) = exp(-\lambda_i^+).(\lambda_i^+)^{\gamma_i}/\gamma_i!$ with expected value, λ_i^+. Given a set of positive$(+)$ and negative examples$(-)$, the relative densities (frequencies normalized by area) of the feature vary in positive and negative images, and can be used to identify crowd patches from non-crowd ones. Assuming independence among features, the log-likelihood $\varphi(P)$ of the ratio of patch containing crowd to non-crowd is [19]:

$$\begin{aligned} &log(\gamma_1, \gamma_2, \ldots \gamma_c|crowd) - log(\gamma_1, \gamma_2, \ldots \gamma_c|\neg crowd) \\ &\qquad = \sum_{i=1}^{c} \big(\lambda_i^- - \lambda_i^+ + \gamma_i(log\lambda_i^+ - log\lambda_i^-)\big). \end{aligned} \tag{3.2}$$

The above equation gives us a confidence for presence of crowd in a patch. The resulting confidence maps are shown in Fig. 3.4 for two images.



-0.10	-0.09	-0.05	-0.06	-0.11	-0.13	-0.14	-0.13
-0.18	-0.14	-0.14	-0.21	-0.12	-0.12	-0.14	-0.14
0.14	0.11	0.18	0.24	0.21	0.24	0.29	0.29
0.20	0.39	0.40	0.31	0.47	0.24	0.39	0.24
0.33	0.38	0.32	0.31	0.32	0.43	0.37	0.20
0.19	0.40	0.31	0.24	0.18	0.29	0.40	0.18
0.24	0.19	0.29	0.20	0.26	0.20	0.08	0.27
0.31	0.12	0.21	0.16	0.07	0.17	0.08	0.03

-0.54	-0.34	-0.26	-0.29	-0.33	-0.35	-0.28	-0.23
-0.07	-0.24	-0.16	-0.15	-0.18	-0.25	-0.15	-0.21
0.11	0.03	0.00	0.03	0.01	0.05	0.06	0.11
0.27	0.20	0.17	0.21	0.36	0.17	0.26	0.13
0.22	0.31	0.27	0.43	0.37	0.44	0.47	0.34
0.20	0.22	0.17	0.27	0.12	0.22	0.35	0.21
-0.01	0.04	-0.06	0.14	0.05	0.01	-0.03	0.02
-0.09	-0.12	-0.00	0.01	-0.02	-0.08	-0.06	-0.11

Figure 3.4: Images with their confidence maps: The images on the left have confidence of crowd likelihood obtained through Eq. 3.2. In the top image, the gap between stadium tiers gets low confidence of crowd presence. Similarly, patches containing the sky and flood lights in bottom image have low probability of crowd.

3.1.2 Fusion of Three Sources

For learning and fusion at the patch level, we densely sample overlapping patches from the training images and using the annotation, obtain counts for the corresponding patches. The features from the three sources are concatenated in an early fusion fashion giving a 20d vector. Next, we scale individual bins of the 20d vector between 0 and 1, and regress using ϵ-SVR, with the counts computed from the annotations.

3.1.3 Counting in Images

In order to impose smoothness among counts from different patches, we place them in an MRF framework with grid structure. Furthermore, although small patches have consistent density, they have fewer repetitions or periods and can easily be affected by low-frequency noise. Larger patches, if they have consistent density, have more people, and therefore more periods and better relevant-to-irrelevant frequency ratio. Moreover, it is difficult to ascertain in advance the right scale for analysis for a particular image. This problem lends itself to a multi-scale MRF, an example of which is shown in Fig. 3.5. The graph can be represented with $(\mathcal{V}, \mathcal{E})$ and $\mathcal{N}$ are the four neighbors at the same level and intermediate nodes that connect a patch to layers above and below it. Note that, this multi-scale MRF is different from other hierarchical models used for images, in that the data term (unary cost) for a patch is evaluated independent of the patches at layers above and below it, whereas in image restoration and stereo, data cost for patch at higher level is computed from layer directly below.

The energy function for the multi-scale MRF is given by:

$$E(\ell) = \sum_{p \in \mathcal{V}} D_p(\ell_p) + \sum_{(p,q) \in \mathcal{N}} V(\ell_p - \ell_q), \tag{3.3}$$

where labeling ℓ assigns a label $\ell_p \in \mathcal{L} = \{0, 1, 2, ..., C_{max}\}$ for every every patch $p \in P$. The data term is quadratic, $D_p(\ell_p) = \lambda(\eta_p - \ell_p)^2$ and smoothness term is truncated quadratic, $V(\ell_p - \ell_q) = \min\left((\ell_p - \ell_q)^2, \tau\right)$.

The graph is inferred using Max-Product/Min-Sum BP on grid structure [97]. At any time t, the message that node p sends to q for a label ℓ_q is given by, $m^t_{p \to q}(\ell_q)$:

$$\min_{\ell_p} \left(V(\ell_p - \ell_q) + D_p(\ell_p) + \sum_{s \in \mathcal{N}_p \backslash q} m^{t-1}_{s \to p}(\ell_p) \right), \tag{3.4}$$

and the belief for a label ℓ_q of node q at time t can be obtained as:

$$b_q^t(\ell_q) = D_q(\ell_q) + \sum_{p \in \mathcal{N}_q} m_{p \to q}^t(\ell_q). \tag{3.5}$$

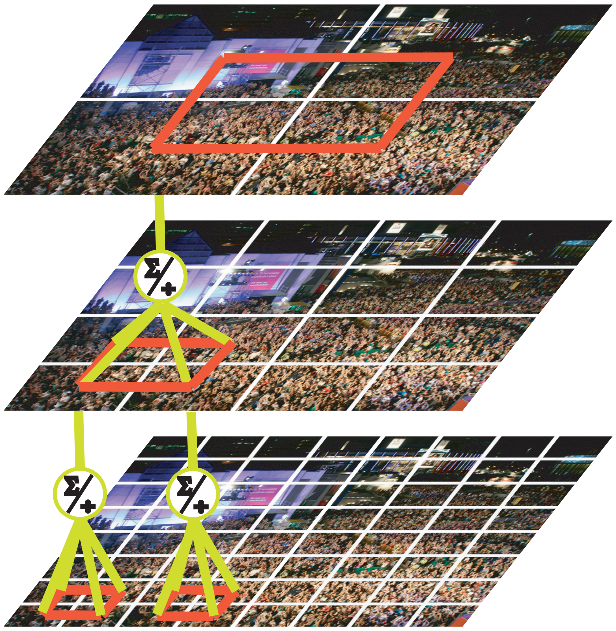


Figure 3.5: The figure shown multi-scale Markov random Field for inferring counts for the entire image. The patches in each layer have independent data terms, thus requiring a simultaneous solution for all layers.

The inference starts by sweeping in four directions at the bottom level using Eq. 3.4, the beliefs are then evaluated for each patch using Eq. 3.5. Then, the beliefs in the groups of 2×2 are added giving the beliefs for the intermediate nodes b_i^t above the bottom layer. After four sweeps at the middle layer, the fifth sweep of messages goes from intermediate nodes to the middle layer. This is followed by computation of beliefs at the middle layer. This step repeats for the top layer, and the whole process corresponds to one time step t. Then, the process repeats but from top to bottom. The beliefs at the intermediate nodes are divided for each of the patch below, i.e., for each patch q in 2×2 group below the intermediate node, its share of beliefs from the layer above is given by: $b_{i,q}^{t+1}(\ell_q) = b_q^t(\ell_q).b_i^{t+1}(\ell_q)/b_i^t(\ell_q)$. After a fixed number of iterations, the final beliefs can be computed using Eq. 3.5, and the labels which have minimum cost in the belief vectors are selected as the final labels. The sum of labels (counts) at the bottom layer gives the count for the image.

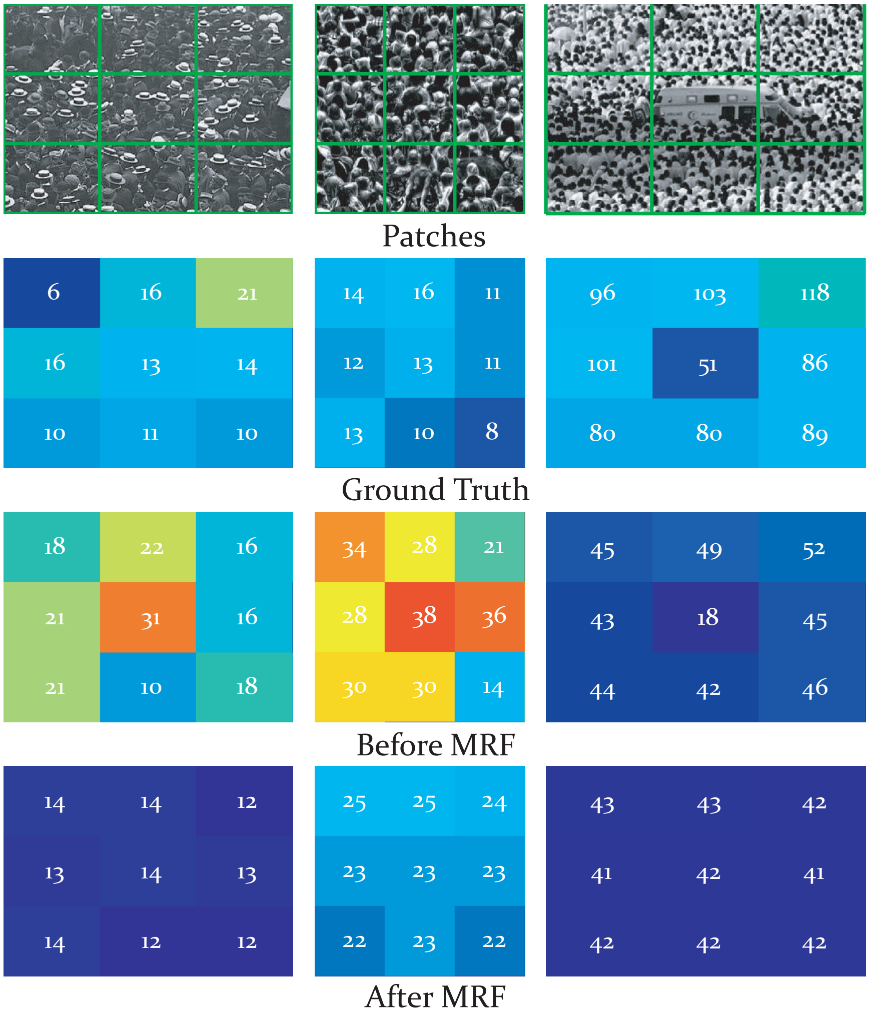


Figure 3.6: Results after MRF-based inference: Three nonets from different images are shown in first row. The second row shows the ground truth counts, and the estimated counts before and after MRF inference are shown in third and fourth rows, respectively. The patches from only one layer are shown in this figure.

Fig. 3.6 shows three instances where the estimated count of patch was improved based on neighbors (both spatial and layer). In all cases, the patch under consideration lies in the center of 3×3 patch set. In the first two columns, after imposing the smoothness constraint using MRF, the overestimated counts get reduced becoming closer to ground truth. A special case is shown in the last column. The patch in the middle had a much lower count than neighbors which after inference increased becoming similar to its neighbors. Although the new estimate is closer to ground truth, the increase is not necessarily correct since the lower count was due to presence of a non-human object (an ambulance). The last column belongs to the image which had the highest count in the dataset.

3.1.4 Experiments

For experiments, we randomly divided the dataset into sets of 10, reduced the maximum dimension to 1024 for computational efficiency, and performed $5-$fold cross-validation. We used two simple measures to quantify the results: mean and deviation of Absolute Difference (AD), and mean and deviation of Normalized Absolute Difference (NAD), which is obtained by dividing the absolute difference with the ground truth count for each image. Since we divide the image into patches, we report our results for both patches and images. The quantitative results are presented in Table 3.1.

The first row in Table 3.1 shows the results of using counts from Fourier Analysis only, giving AD of 703.9 and NAD of 84.6. Supplementing it with confidences from various sources including Eq. 3.2 improves AD by 181.8 and reduces NAD by almost one-half. Including counts from head detections improves AD marginally to 510.9. Adding counts from regression on sparse SIFT features reduces error in both measures, giving values of 468.0 and 32.2, respectively. Finally, inferring counts for complete images using counts from patches through multi-scale MRF further improves AD taking it to 419.5. It can be observed from the table, that standard deviation follows the same trend as mean, the values reducing as we add more sources.

Table 3.1: Quantitative results of the proposed approach and comparison with Rodriguez et al. [17] and Lempitsky and Zisserman [52] using mean and standard deviation of Absolute Difference (absolute error) and Normalized Absolute Difference (percentage error) from ground truth. The influence of the individual sources is also quantified. The proposed approach outperforms the other two methods.

Method	Error			
	Per Patch		Per Image	
	AD	NAD	AD	NAD
Fourier	13.8 ± 21.3	96.4 ± 200.4	703.9 ± 682.0	84.6 ± 157.3
F+**c**onfidence	11.0 ± 19.7	58.7 ± 74.9	522.1 ± 610.1	41.0 ± 31.0
Fc+**H**ead	11.1 ± 19.3	63.3.0 ± 84.0	510.9 ± 587.3	41.8 ± 30.9
FHc+**S**IFT	10.2 ± 18.9	53.3.0 ± 69.5	468.0 ± 590.3	32.2 ± 27.1
FHSc+**M**RF **(Proposed)**	-	-	**419.5 ± 541.6**	**31.3 ± 27.1**
Rodriguez et al.	-	-	655.7 ± 697.8	70.6 ± 102.1
Lempitsky et al.	-	-	493.4 ± 487.1	61.2 ± 91.6

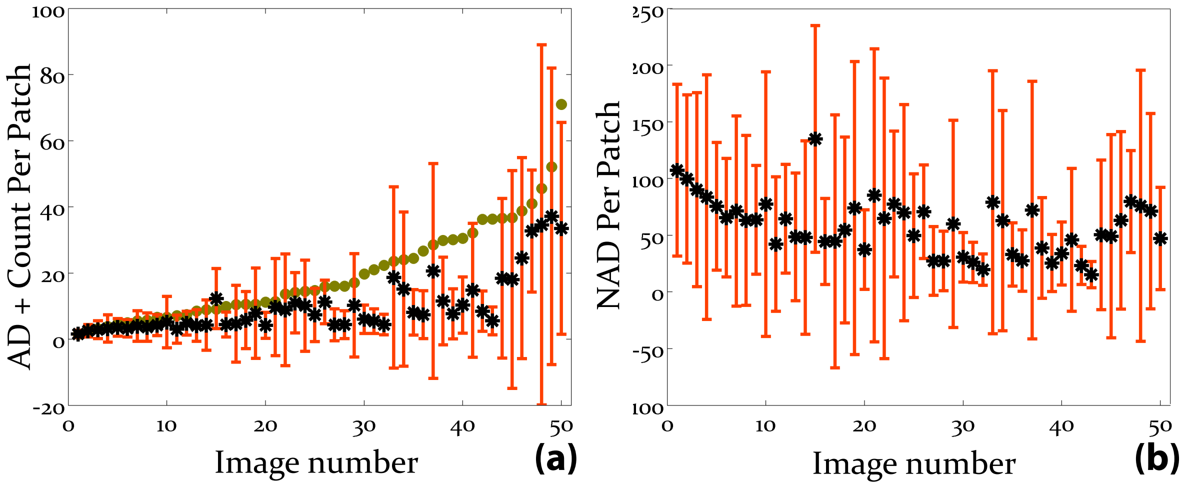


Figure 3.7: This figure shows analysis of patch estimates in terms of absolute and normalized absolute differences. The x-axis shows image number sorted with respect to actual count. Means are shown in black asterisk, standard deviations with red bars, and ground truth counts with olive dots.

Figs. 3.7a-b shows AD and NAD for patches in the individual images, respectively. The mean per patch are shown with black asterisks, deviations with red bars, and olive dots in Fig. 3.7a show average of actual counts per patch in that image. For easier analysis, the x-axis shows images sorted with respect to actual counts in both plots. It can be seen that AD per patch increases as the actual counts increases, except for the images in the range 25 to 45 with corresponding actual counts in the range of $1000 - 2500$ per image. Not only does this range boast lowest mean in AD and NAD, but lowest deviations as well, which means the approach consistently predict correct counts for patches in this range. The reason for better performance in the middle range is obvious: the counts range from $94 - 4543$, so the largest count is a tremendous 4832% of the smallest count. Forcing the learning algorithm to predict correct estimates at both ends simultaneously, makes it overestimate the lower end and underestimate the higher end, thereby working in favor of the middle range, even though, we used RBF kernel for regression on three sources.

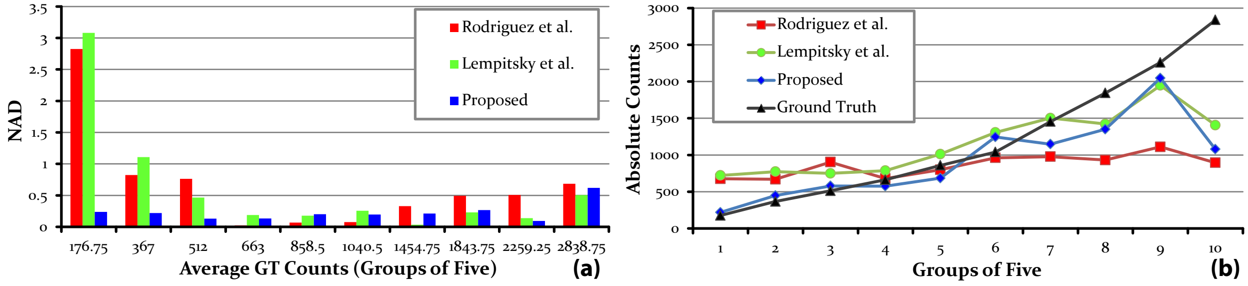


Figure 3.8: Analysis of comparison: Bars and lines in red depict [17], green show [52], blue shows the results using proposed approach, while ground truth is shown in black. (a) shows Normalized Absolute Difference (an error measure) and (b) shows the actual and estimated counts.

For comparison, we used the methods of Rodriguez et al. [17], and Lempitsky and Zisserman [52], which were suitable for this dataset since other methods for crowd counting mostly deal with videos or use human detection, and cannot be used for testing on this dataset. The method presented in [17] relies on head detections, while [52] requires annotated ground truth points for training, and learns a regression model using dense SIFT features on randomly selected patches. The quantitative results are shown in Table 3.1. Fig. 3.8 breaks these numbers according to counts.

The results using [17] are in red, those in green use [52], and the results of the proposed approach are shown blue. In Fig. 3.8b, the black curve represents the ground truth. In Fig. 3.8a, we show NAD for ten groups of five images each, which are sorted according to ground truth counts. The x-axis shows the average counts of each of the 10 groups. Density aware person detection [17] performs best around counts of 1000, but its error increases as we move away. The reason becomes obvious when we look at the absolute counts output by the method in Fig. 3.8b, as they are fairly steady across the entire dataset and do not respond well to change in density. It overestimates at lower end and then underestimates at the higher end, resulting in increased absolute errors on both ends. The MESA-distance [52] on the other hand, performs fairly well at higher counts, but gives high NAD at lower counts. The reason lies in the algorithm itself, as it is designed to minimize the maximum AD across images when training, and since images with higher counts tend to have higher AD, the learning focuses on such images. The learner gets biased towards high density images, thus, producing a lower AD overall, but overestimating at lower counts (Fig. 3.8b), thus giving higher NAD. The proposed approach, on the other hand, performs well across the whole range, giving steady NAD's across all ten groups.

Finally, all methods underestimate the tenth set and this can be due to several reasons. First, images in this group are very high resolution and therefore it is less likely to miss individuals while annotating. Since we fixed the maximum image size for experiments, the images in this group had correct and therefore, more annotations than their low-resolution counterparts. Second, a careful look at Fig. 3.7a reveals that patch density increases super-linearly for this group, which otherwise is linear for first nine groups. Since there are few such images, their patch instances could have been treated as outliers (have higher slack weights) for regression. The last reason may be associated with histograms of features that capture relative frequencies. At very high density, the relative frequencies across patches with different density may become similar, resulting in a loss of discriminative power.

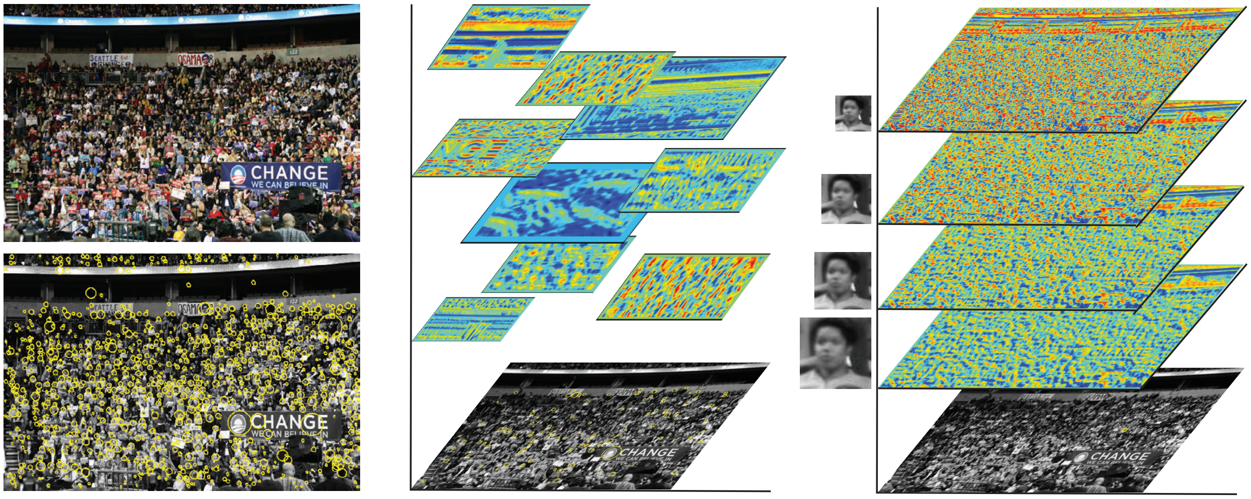


Figure 3.9: This figure shows the schematic outline of the proposed method for localization. (Left) Given an image on lop-left, we initialize our algorithm using an underlying head detector (bottom-left). (Middle) Next, we select high-confidence hypotheses within each cell in a pre-defined grid and a fixed number of high-confidence hypotheses from the entire image, and correlate them across their neighborhoods. (Right) Another set of detections is selected representing most frequent yet discriminative hypotheses which are correlated across the entire image at multiple scales.

3.2 Localization in Dense Crowds

The problem of localization is significantly more challenging than counting because instead of producing a single estimate of count for the entire image, we have to pinpoint each individual for the purpose of localization. For that, we exploit the contextual yet redundant information available in an image of extremely dense crowd. We begin by discovering recurring patterns in the image using initial head detection hypotheses output by a detector. When the head detector trained on an independent training data is applied on an image of a dense crowd, the recall is very limited and only a subset of true detections are output. The reason lies with severe occlusions, differences in illumination and pose, and more frequently with small head sizes for regions in the image capturing distant locations from the camera. The initial detections serve as modes or atoms of the pattern that we wish to discover. We used the modified Deformable Parts Model [23] similar to our approach for counting and use only the template corresponding to head for detection. Other parts, being mostly invisible, are not attempted for detection. Figure 3.9 describes the outline of the approach.

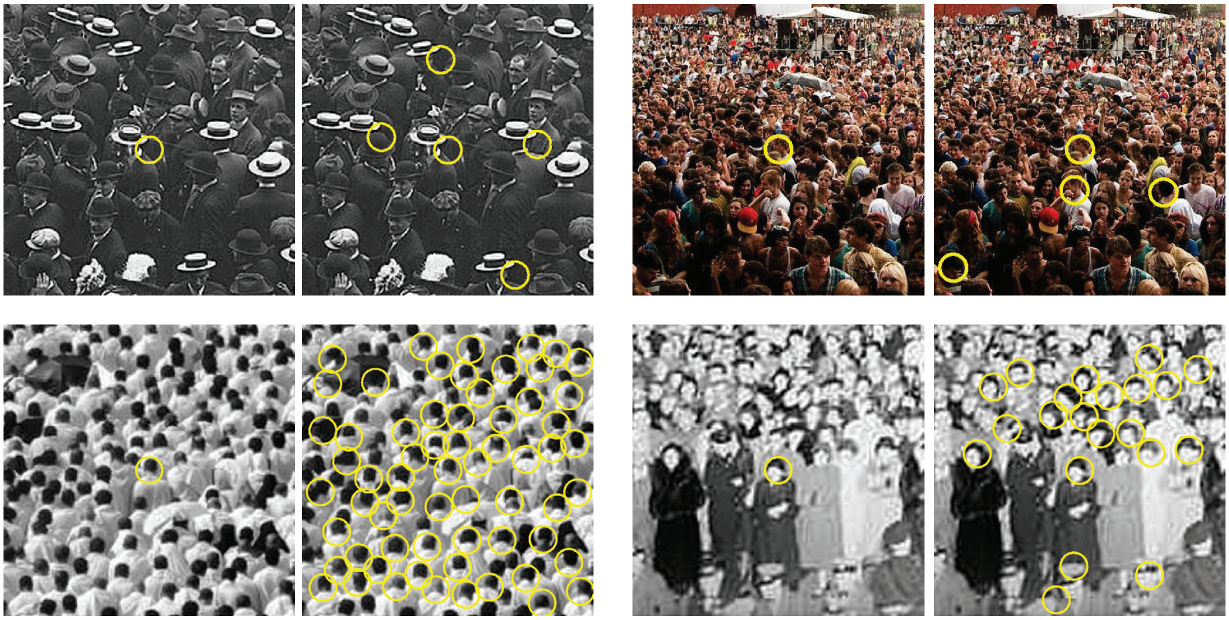


Figure 3.10: This figure shows the results of pattern matching within neighborhoods of high-confidence detections. In the first and third columns are the original patches, while the second and fourth columns show the new hypotheses discovered through correlation. Interestingly, in the top-left the original patch corresponded to only a part of a hat, which was then re-discovered at many other places.

3.2.1 Finding Repetitive Patterns

Given an initial set of detection hypotheses, we find other regions in the image which are similar in appearance. However, not all hypotheses are correct and searching for meaningless patches corresponding to false positives will only lead to deteriorated performance. Therefore, only the high-confidence detections are selected for finding repetitive patterns. For that, the patch corresponding to the detection is correlated within its neighborhood, followed by non-maximal suppression to reduce overlap. The confidence for the new hypotheses is obtained as the product of correlation score with the score of the original detection. Figure 3.10 shows the output of this procedure on four patches takes from different images. The top-left pair in the figure shows localized parts of hats instead of heads. Although this may introduce error in the localization in terms of distance from the true location, it is still useful since it allows localization of more individuals than is possible with only detection hypotheses output by the head detector.

Furthermore, the detections from the region closer to the cameras are more likely to be detected with high confidence. To ensure that repetitions are discovered from all regions of the image, we divide it into fixed size grid cells and correlate the high-confidence detections in each cell across the entire cell. Intermediate results are shown in Figure 3.11.

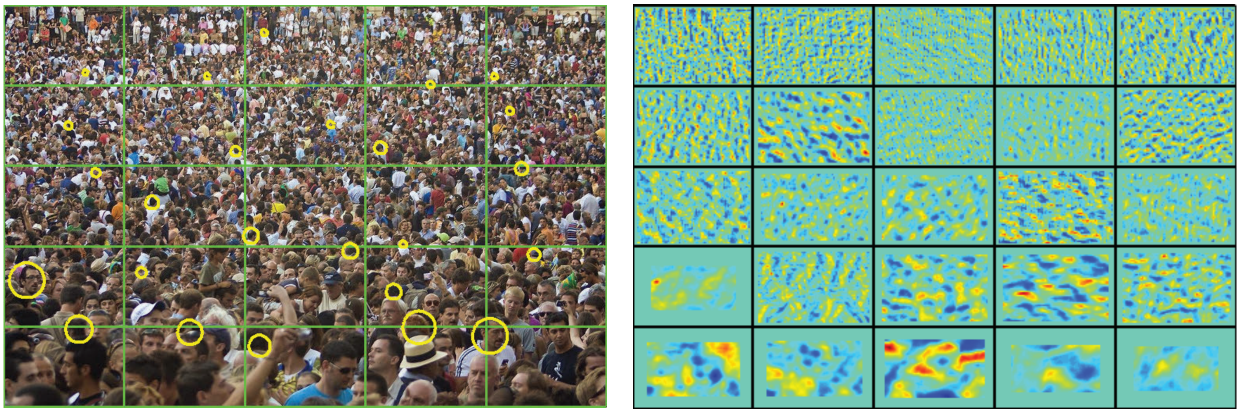


Figure 3.11: Results of hypotheses expansion within grid cells: A given image is divided into a pre-defined number of cells as shown on the left. In each cell, the highest confidence detection hypothesis is selected and correlated across the cell as shown on right. This ensures that the repetitive patterns are discovered in the entire image.

Both of previous criteria are applied locally since the change in scale due to perspective distortion makes the correlation meaningless at scales different from the original hypotheses. Similarly, the illumination and pose of head for a particular detection will also share similarity only in its local neighborhood. To find a set detection patches that can be searched in an entire image at multiple scales, we resize the templates for head patches to a fixed size (40×40) and then convert them to 1d vectors. Next, we cluster all the vectors using k-means. Then, using a weighted combination of discrimination power, confidence and distance from the cluster mean, we select a fixed number of hypotheses for correlation across the image. Figure 3.12 shows the results where the data was grouped into 10 clusters. For each cluster, we show the cluster center in yellow, and the three most representative detection patches in red, green and blue, respectively.

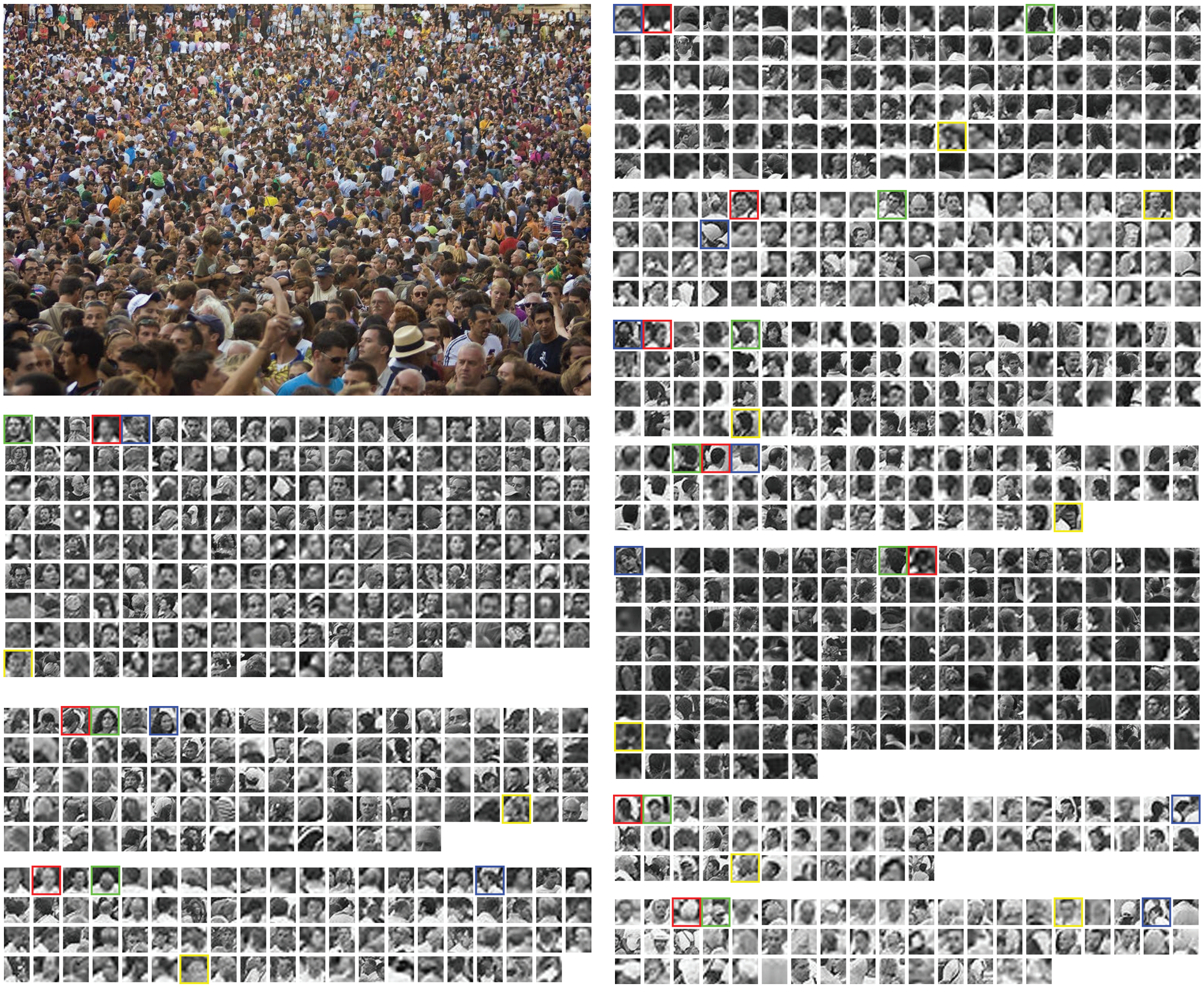


Figure 3.12: This figure shows the intermediate steps for discovering representative detection hypotheses. Each group of patches represents one cluster where the cluster center (mean) is marked with yellow box, while the top three representative patches are shown with red, green and blue, respectively.

3.2.2 Hypotheses Pruning under Scale Constraints

The approach presented in the previous section provides an over-complete set of hypotheses, i.e., at each location there are multiple competing hypotheses possibly at different scales. Many of these hypotheses are redundant or wrong, and, therefore need to be eliminated. This is a variable selection problem which can be solved with binary integer programming that we present in this section.

Let z_q denote the binary variable associated with the detection hypothesis q where $q \in \{1, 2, 3, ..., Q\}$, and x_q, s_q and c_q, denote its location, scale and confidence, respectively. Furthermore, let the template around x_q be given by T_q. Our goal is then to maximize the number of selected hypotheses with selection preferences determined by the confidences, i.e., a hypothesis with a higher confidence should have a higher chance of selection. Since, dense crowd is repetitive, the selected hypotheses should resemble their respective neighbors, both in terms of scale and appearance. Finally, at each location only a single person can be present, therefore, all other competing hypothesis must be deselected by the optimization. Thus, our objective function becomes the following:

$$\begin{aligned} \min_{z_q} \quad & -\sum_q z_q c_q && + \lambda_\Gamma \sum_q \sum_{q' \in NN_q} z_q z_{q'} \Gamma\big(s_q, s_{q'}\big) \\ & && + \lambda_\Omega \sum_q \sum_{q' \in NN_q} z_q z_{q'} \Omega\big(T_q, T_{q'}\big), \qquad (3.6) \\ \text{s.t.} \quad & z_q && + \sum_{q'' \mid (x_{q''} - x_q)^2 \leq s_q} z_{q''} = 1, \qquad \forall q \in \{1, 2, 3, ..., Q\}, \qquad (3.7) \\ & z_q \in \{0, 1\}. && \qquad (3.8) \end{aligned}$$

where Γ and Ω compute the dissimilarity in scales and appearance of two hypothesis, and λ_Γ and λ_Ω are the corresponding weights, respectively. We define $\Gamma(s_i, s_j) = \big(s_i - s_j\big)^2$ and $\Omega(T_i, T_j) = \sum_{\text{pixels}} \big(T_i - T_j\big)^2$. The constraints in Equation 3.7 are known as Special Order Set (SOS) Type

1 constraints (Figure 3.13) and state that at each location occupied by a particular hypothesis, at most a single hypothesis can be selected by the optimization which can be that hypothesis or any of the competing hypotheses.

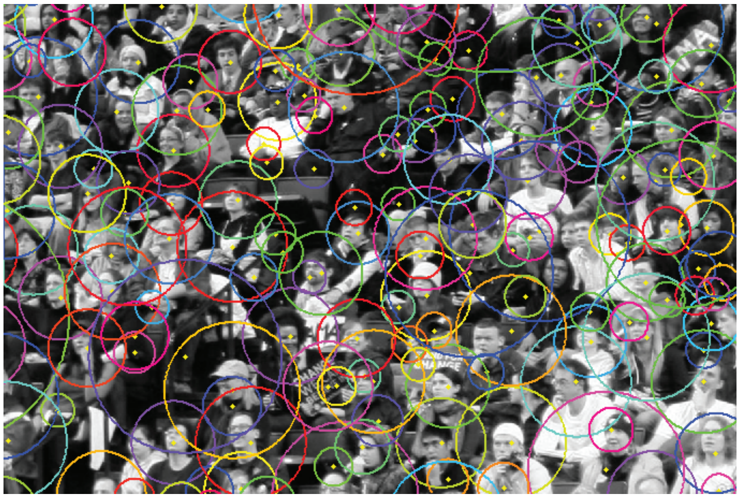


Figure 3.13: This image shows the Special Ordered Set (SOS) Type 1 constraints that are used during final selection of hypotheses. Each hypothesis (shown with yellow dot) enforces a constrain in its neighborhood (shown with colored circle) so that only one hypothesis is output by the algorithm at that location.

The constrained minimization in Equations 3.6-3.8 can be solved through binary integer quadratic programming subject to SOS constraints, given by;

$$\begin{aligned} \min \quad & z^T \mathbf{Q} z + \mathbf{f}^T z, \\ \text{s.t.} \quad & \text{SOS constraints}, \\ & z \in \{0,1\}^Q. \end{aligned} \tag{3.9}$$

where $\mathbf{Q}(i,j) = \lambda_\Gamma \Gamma(s_i, s_j) + \lambda_\Omega \Omega(T_i, T_j)$ and $\mathbf{f}(i) = -c(i)$. The SOS constraints are obtained from Equation 3.7.

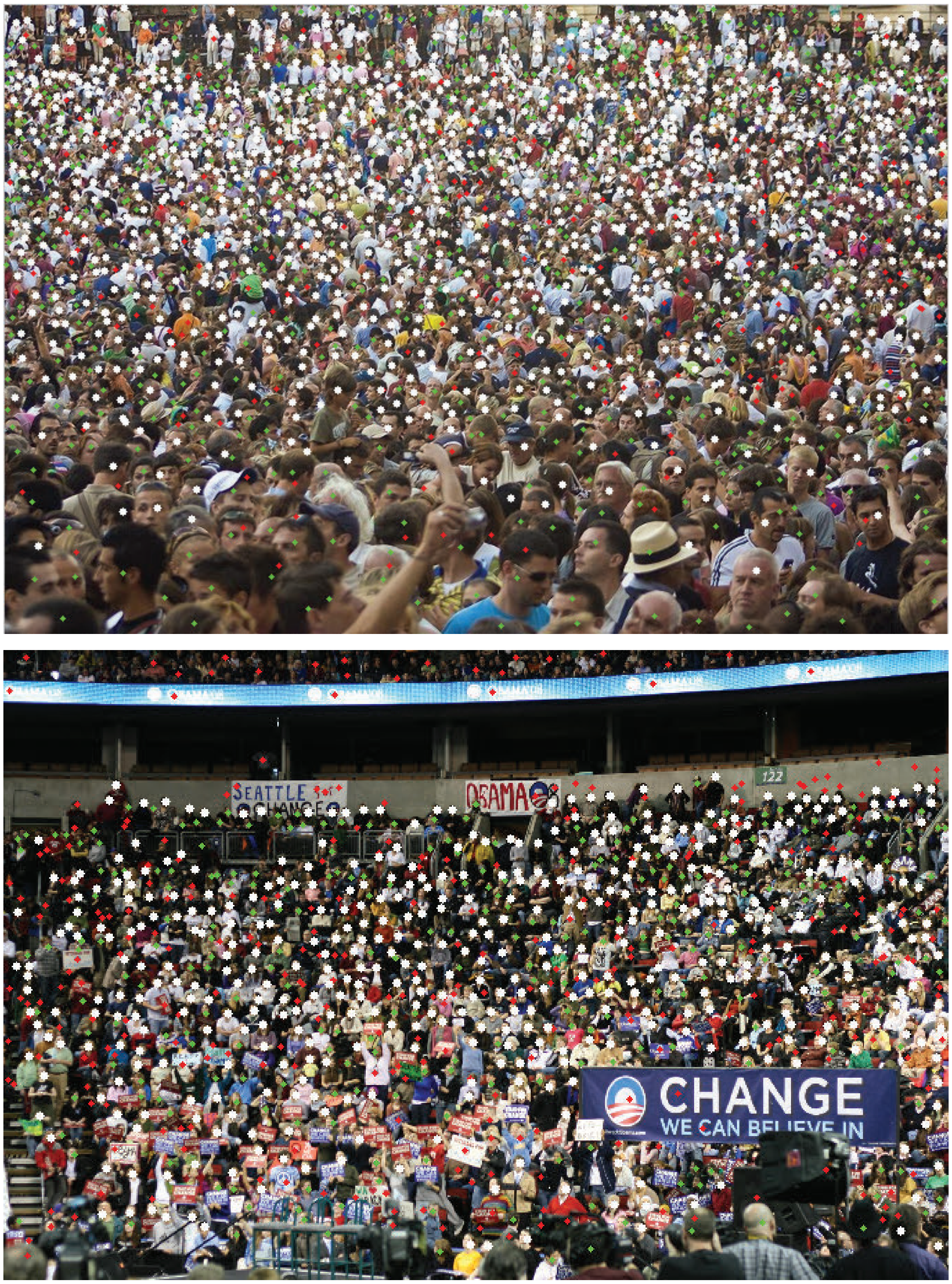


Figure 3.14: This image shows the final results of localization approach. Here, white dots signify true positives (correct localization), green show false negatives (miss detections) while red dots indicate false positives.

3.2.3 Experiments

We performed experiments on the same dataset introduced for counting in the previous section. Since the dataset has dotted annotations that mark the location of each individual, it becomes possible to evaluate the performance of the localization method. In Figure 4.10, we show qualitative results for two images, where white dots show correct localizations, while green and red dots indicate the errors. We quantify the results using Average Precision where different values of precision and recall are obtained by changing the distance threshold between ground truth and the output detection location. Detections within the threshold distance are the True Positives. On average, the proposed approach improves the Average Precision (AP) over the baseline head detector by almost 3%.

3.3 Chapter Summary

In this chapter, we presented two approaches for counting and localizing individuals in images of extremely dense crowds containing on average a thousand people. An analysis of crowds on such a scale has not been tackled before in literature. We fuse information from three sources in terms of counts, confidences and different measures at the patch level, and then enforce smoothness constraint on nearby patches to improve estimates of incorrect patches, thereby producing better estimates at the image level. For the task of localization on images of same dataset, we proposed to find repetitive patterns with initializations provided by head detections. This procedure increases the recall of detections which is otherwise restricted and dependent on the recall of underlying head detector. Finally, we proposed to filter out expanded hypotheses by posing the problem as a binary integer quadratic programming problem with Special Order Set Type 1 constraints. Experimental evaluation for both approaches shows promise for their effectiveness in analysis of images containing people on the order of thousands.

CHAPTER 4: HUMAN DETECTION USING LOCALLY-CONSISTENT SCALE PRIOR AND GLOBAL OCCLUSION REASONING

In this chapter, we describe in detail our approach for human detection in images of dense crowds. To keep the chapter self-contained, we first describe essential details of Deformable Parts Model [24] that are relevant to our approach, using the same notation as in [24]. We, then, describe how scale and confidence priors are automatically discovered from given images by refining the priors and human detections in an iterative fashion. Next, we present a technique to detect combinations-of-parts using the existing DPM formulation. Finally, the set of putative detections are globally reasoned for occlusion, resulting in bounding boxes on the visible parts of humans as output. Note that the choice of DPM as underlying detector is arbitrary, any human detection algorithm which performs detection at multiple scales and uses part-based models can be substituted in its place.

4.1 Background: Deformable Parts Model (DPM)

In order to capture the changes in viewpoint as well as variations in pose due to the articulation, Deformable Parts Model [24] uses HOG features to match appearance, and instead of using just the filter scores from rigid templates, it considers deformation, which when represented as a score, measures the displacement of parts from their ideal locations.

To detect objects at multiple scales, a feature pyramid H is constructed with L levels, with $p = (x, y, l)$ representing the position (x, y) at level l in the pyramid. The parameter λ determines the rate for scale sampling in H, i.e., λ is the number of levels down the pyramid at which the resolution doubles compared to a given level. The feature vector at position p in the pyramid is given by $\phi(H, p)$. The appearance score is, then, simply the dot product between filter, F', and feature vector, i.e., $F' \cdot \phi(H, p)$. The model for part i is represented as $P_i = (F_i, v_i, d_i)$, where F_i is the filter for the i-th part, v_i is the anchor position w.r.t root position, and d_i is the deformation cost. The deformation score of a part with displacement (d_x, d_y) is given as $d_i \cdot \phi_d(d_x, d_y)$, where ϕ_d returns the deformation features. Finally, an object model with n parts is given by $(F_0, P_1, P_2, ...P_n, b)$, where F_0 is the root filter, P_i is the model for ith part consisting of appearance and deformation costs, and b is the constant bias term. The confidence output by human detector, conf_{HD}, for each hypothesis is the sum of scores from the root filter, filter and deformation scores from the parts, plus the bias, i.e.,

$$\text{conf}_{\text{HD}}(p_0, p_1, p_2, ...p_n) = \sum_{i=0}^{n} F'_i \cdot \phi(H, p_i) - \sum_{i=1}^{n} d_i \cdot \phi_d(d_{x_i}, d_{y_i}) + b. \tag{4.1}$$

4.2 The Scale and Confidence Priors

In a densely crowded image or video, human detection becomes difficult primarily due to the smaller target size and severe occlusions. But, the scale of a human in crowded scene provides cue to what the scale should be in the immediate surrounding of the associated detection. We can transfer the knowledge of scale from a point in scene to its surroundings using the scale and confidence of that particular human detection. Figure 4.1 illustrates this idea. Given scale and confidence priors, the confidence for detection hypotheses is altered to reflect conformity with the priors. However, since both the priors and detections are dependent on each other, this necessitates an iterative mechanism where the priors are improved using given detections, and detections are improved using updated priors. Next, we present one cycle of this iterative procedure to discover priors and obtain detections.

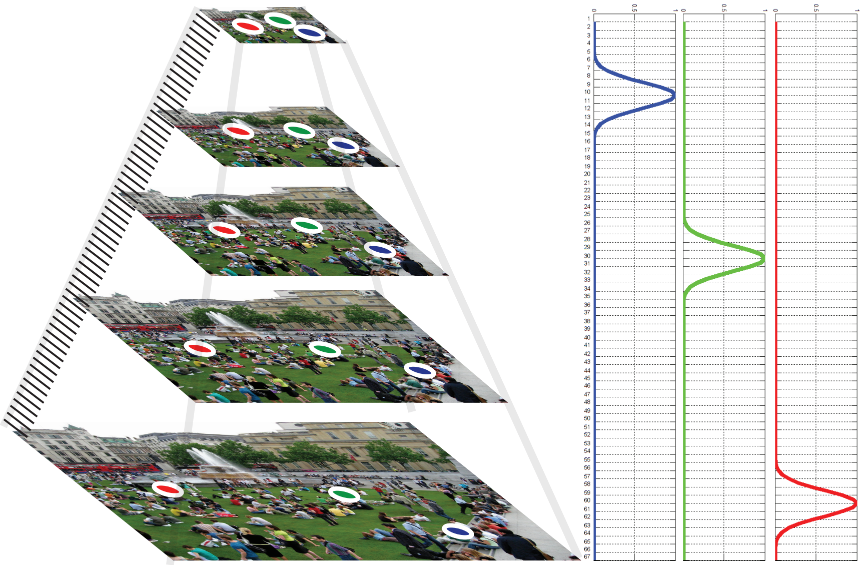


Figure 4.1: Human detection in DPM [24] is performed using $L = 67$ levels of the pyramid. Three pixel locations in given image, shown with different colors, have different prior information on scale and confidence. In our approach, the scale and confidence priors are discovered automatically which then provide a 1d scoring function at each pixel in the image, as shown on right. By transforming the priors to each level of the pyramid, the confidence for detection hypotheses is altered based on their consistency with the priors. Increasing the confidence of scale-consistent but low-confidence hypotheses allows them to be detected without incurring false positives in the rest of the image. Effectively, for a 2304×3072 image, this amounts to re-scoring all the 3.85 million hypotheses.

Inferring scale and confidence priors from given detections: For a detection Ω_q, let (x_q, y_q) denote its position in the image, and s_q and c_q represent the scale and confidence, respectively. Then, given a set of input detections, $\Omega_q, q = 1, 2, ..., Q$, our goal is to infer the scale and confidence at each location $\mathrm{x} = (x, y)$ in the image. All the detections induce a local influence in terms of scale and confidence, which can be captured with an *Influence Function*, induced by every detection. Such a function should be dependent on locations of input detections since scale-consistency is only valid locally in most images depending on camera location, number of ground planes (stairs, stadiums) or number of head planes (people sitting, standing). It should be a function of scales because a detection with a larger scale has its neighbors at a larger distance than smaller detections. Finally, since the scale information of high-confidence detections is more reliable, it should also be dependent on the confidence, c_q. We propose to use the following function,

$$\xi_{x,y}(\Omega_q) = c_q \cdot \exp\left(-\frac{\|x - x_q\|^2 + \|y - y_q\|^2}{\sigma^2 \cdot (1 + s_q/\rho(H_{L/2})^2}\right), \tag{4.2}$$

where σ is the deviation along x and y axes, and $\rho(H_l)$ returns the scale of a detection at level l in the pyramid.

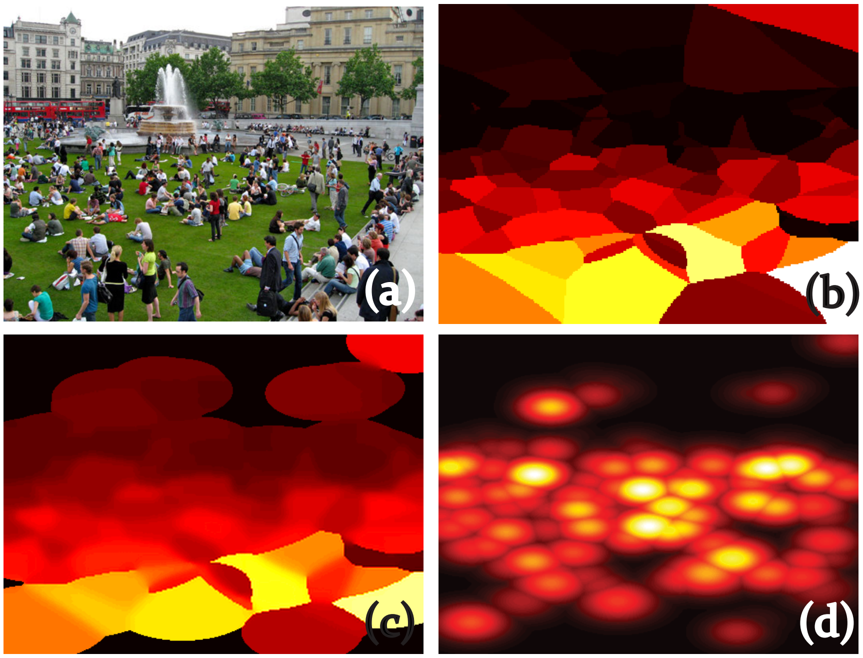


Figure 4.2: Intermediate computations of scale and confidence priors: (a) The scales and confidences from detections in an image are transformed into a 2d graph. (b) The observed scale prior is obtained using Equation 4.3, (c) which is then smoothed through MRF using Equation 4.4. The corresponding confidence prior is also shown in (d). Heat map is used in (b)-(d) where brighter colors indicate larger values.

From Equation 4.2, it is evident that $\xi_{x,y}$ is a function of all three aspects of a detection, its location (x_q, y_q), scale s_q and confidence c_q. It also satisfies all the mentioned properties and, therefore, is a valid Influence Function. Furthermore, the detection that has the maximum value at the location (x, y) in the image, Ω_{q*}, determines the value of scale (Θ_s) and confidence (Θ_c) priors at that location, i.e.,

$$q* = \underset{q}{\operatorname{argmax}}\, \xi_{x,y}(\Omega_q), \forall q = 1, 2, 3, ..., Q, \\ \Theta_c(x, y) = \xi_{x,y}(\Omega_{q*}), \quad \Theta_s(x, y) = s_{q*}. \tag{4.3}$$

The confidence prior Θ_c at each location in the image is just the maximum value of Influence Function. The scale prior Θ_s is the scale of the particular detection that has the maximum influence at that location. Figure 4.2(b) shows the scale prior for an image shown in Figure 4.2(a). It is similar to Voronoi Diagram except each region is represented by a scale and the distance-measure is the influence function ξ, instead of the Euclidean distance.

Due to perspective effects, the scale of humans changes from pixel to pixel, but its affect is usually gradual. While the humans closer to the camera appear larger, the ones in the background appear smaller. This consistency in scale is imposed by treating scales as random variables and placing them in a Markov Random Field which enforces smoothness at nearby image locations. We model this using grid MRF (inferred using Max-Product/Min-Sum BP [97]), and is given by:

$$E(\ell) = \sum_{\text{x} \in \mathcal{V}} \Phi_{\text{x}}(\ell_{\text{x}}) + \sum_{(\text{x},\text{x'}) \in \mathcal{N}} \Psi(\ell_{\text{x}} - \ell_{\text{x'}}), \tag{4.4}$$

where Φ, Ψ are the unary and binary potentials and $\mathcal{V}, \mathcal{N}$ define vertices (pixel locations) and neighborhoods in the graph, respectively. The labeling ℓ assigns a label (scale) at every location x in the image. The data term, Φ_{x}, is quadratic, while smoothness term, Ψ, is truncated quadratic:

$$\Phi_{\text{x}}(\ell_{\text{x}}) = \eta(s_{\text{x}} - \ell_{\text{x}})^2, \\ \Psi(\ell_{\text{x}} - \ell_{\text{x'}}) = \min\left((\ell_{\text{x}} - \ell_{\text{x'}})^2, \tau\right).$$

Although the scale varies gradually due to perspective effects, but due to particular viewpoints, there can exist sharp discontinuities. These can also arise from false positives which are likely to be different in scale than correct detections in a particular neighborhood. Thus, it is important to infer the scale prior while preserving the sharp discontinuities. The truncated quadratic cost for smoothness allows us to achieve this objective. Figure 4.3(a) and (c) show the case of rapid scale change and that of scale-inconsistent false positives, respectively. In both cases, the scale information was correctly captured using the proposed approach. The yellow arrow in Figure 4.3(a) marks the gradual change in scale around the pond, while the yellow line placed on the pond marks rapid change in scale across it. The humans near the camera have a larger size, while those behind the pond are much smaller. The same effect is shown in the inferred scales shown in (b). Similarly, in (c), the initial false positive on the traffic light, shown with yellow square, was not allowed to corrupt nearby correct scales as it was larger in size than other detections.

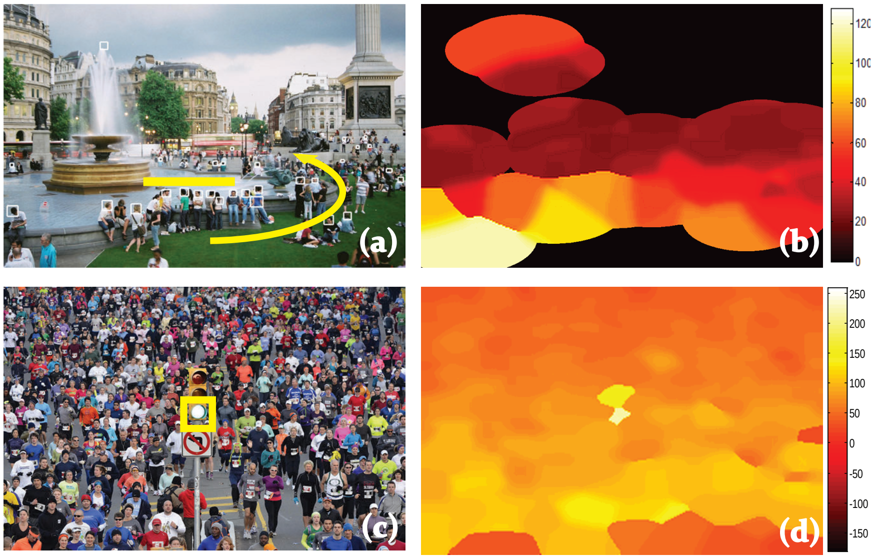


Figure 4.3: Intermediate results on inferred scales after smoothing: Two images are shown in (a) and (c) while the inferred scale priors are shown in (b) and (d), respectively. Truncated quadratic cost in Equation 4.4, allows us to handle sharp discontinuities in the scale field, likely to happen at specific viewpoints and due to false positives. The image in (a) has a fountain, where there is a gradual change in scale around it (yellow arrow) but a sharp discontinuity across it (yellow bar). Similarly, in (c), the initial set of detections had a false positive at traffic light larger in size than the immediate neighbors. In both cases, the gradual change in scale and discontinuities were preserved by MRF.

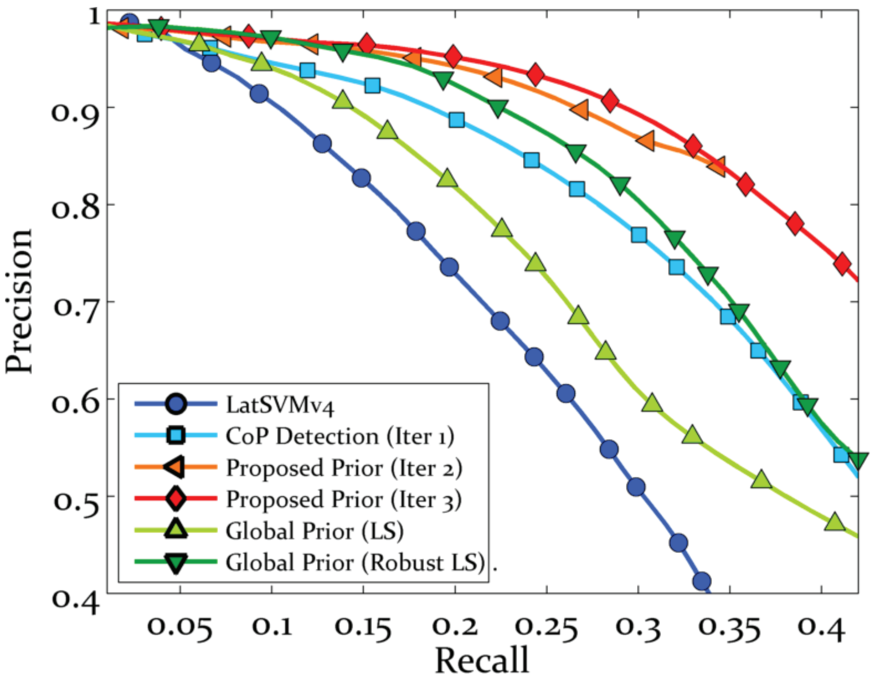


Figure 4.4: This graph shows the improvement in performance obtained using the priors over three iterations. The y and x axes show precision and recall, respectively. The curve without the priors is in blue, while the curve with the priors after iterations is in red. Also shown are the results of global scale priors in greens. Details of experimental setup are in Sec. 5.5.

Altering confidences of detection hypotheses given priors: Given priors, the confidences of detection hypotheses are then re-evaluated, as illustrated in Fig. 4.1. The new confidence is the sum of confidence from the underlying human detector plus the output of scoring function that measures consistency of scale of the detection hypothesis with the scale prior at that location weighed by the confidence prior,

$$\text{conf}(\Omega_q) = \text{conf}_{\text{HD}}(\Omega_q) + \alpha \cdot \Theta_c(x_q, y_q) \cdot \exp\left(-\frac{1}{\beta}\|s_q - \Theta_s(x_q, y_q)\|^2\right), \tag{4.5}$$

where α, β are the parameters of the scoring function.

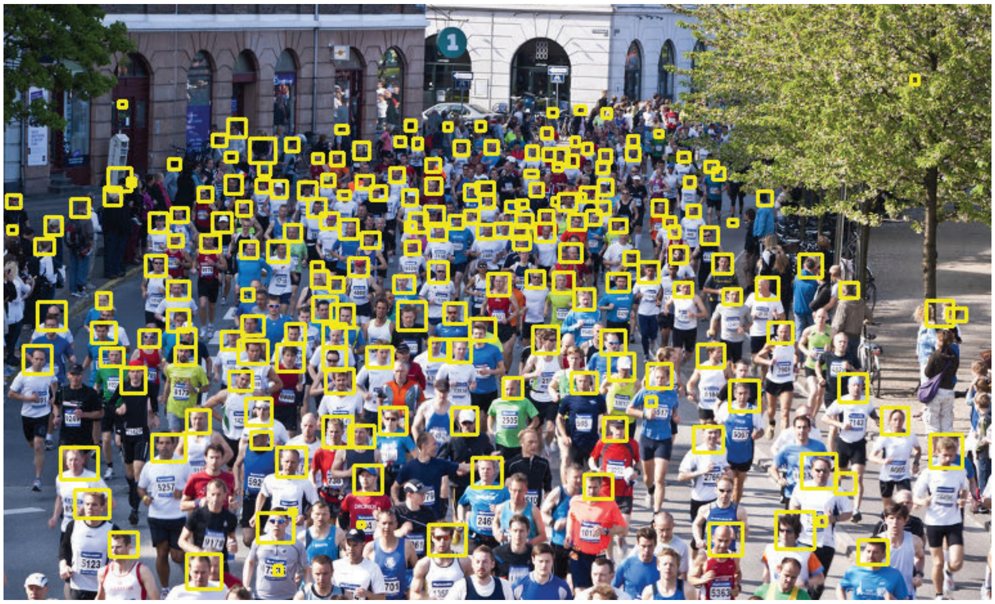


Figure 4.5: This figure shows results after using scale-consistency and combinations-of-parts detection on an image.

Transformations between priors and feature pyramid: From implementation's perspective, there are two important transformations between scale and confidence priors and each level in the feature pyramid. The first relates the x and y coordinates in priors, (x_Θ, y_Θ), which are the same size as the image, to those in level l in the pyramid, (x_{H_l}, y_{H_l}), and is given by:

$$\begin{bmatrix} x_{H_l} \\ y_{H_l} \end{bmatrix} = \begin{bmatrix} \frac{\rho(H_l)}{\rho(H_{l_0})\cdot k} & 0 & w_0+1 \\ 0 & \frac{\rho(H_l)}{\rho(H_{l_0})\cdot k} & h_0+1 \end{bmatrix} \begin{bmatrix} x_\Theta \\ y_\Theta \\ 1 \end{bmatrix}, \tag{4.6}$$

where k is the block size used for constructing HOG, w_0, h_0 are the width and height of root filter F_0, and $\rho(H_l)$ is the scale at level l. The second transformation relates the scale in the image or priors to that of each level in the feature pyramid. The $1-1$ mapping that relates size of detection (root template) at image/prior scale to the level l in the pyramid is given by:

$$s = \frac{w_o \cdot h_0 \cdot k}{\rho(H_l)/\rho(H_{l_0})} - 1. \tag{4.7}$$

In case, we desire to measure the scale of detections in terms of some specific part instead of the root template, for instance the one corresponding to head, we can replace w_0, h_0 with dimensions of that filter in Equation 4.7, and in image space compute the area of bounding box associated with that part. However, since responses to part filters are computed at twice the resolution, H_l must be replaced with $H_{l+\lambda}$.

The above procedure describes details for one iteration of inferring priors and detecting humans. Figure 4.4 quantifies the improvement obtained by using the priors at each iteration, as well as the results of baseline [24] and combinations-of-parts detection which is presented in next section. The results are evaluated using only the heads to discard the effect of bounding box sizes. There is very little improvement after third iteration, so we used 3 iterations for our experiments. The results improve over iterations because 1) the correct detections typically have high confidence, and therefore, more influence on their surroundings, 2) the scoring function increases confidence of only those detection hypotheses that are consistent with the scale prior which is smoothed and inferred using many detections from the previous iteration. 3) Although the detections are pre-processed to remove outliers (median filtering) in terms of scale, even if some scale-inconsistent false positive gets through, the discontinuity-preserving MRF ensures that its impact remains restricted. In addition, we report results of global head plane estimation with least squares (LS) in light green and robust LS using RANSAC (RLS) in dark green. Robust LS improves results over baseline but proposed method still outperforms either method of head plane estimation, primarily due to violation of single ground / head plane assumption in many images and re-scoring of all hypotheses before they are selected for output. In Figure 4.5, we show how the priors yield excellent results in an image containing a dense crowd. For clarity, all detections are visualized using only heads which are drawn in yellow.

4.3 Combination-of-Parts (CoP) Detection

Since a human detector always looks for a complete human, it yields low confidences for individuals who are partially visible. To detect such occluded humans in the image, we can lower the threshold, but that incurs false positives which may have higher confidences than the correct but partially visible humans. And since we are dealing with crowded images characterized by severe occlusions, this phenomenon becomes significant. The solution is to detect multiple combinations of parts, which depending on the visibility of parts of an individual, will correspondingly give higher confidence detections. For our approach, we use four different combinations: head $\mathbb{C}_h$, head and shoulders $\mathbb{C}_s$, upper body $\mathbb{C}_u$ and full body $\mathbb{C}_f$. We modified the DPM implementation to detect different combinations of parts by ignoring the filter and deformation scores of excluded parts in each combination. Excluding certain parts affects the Latent SVM bias, since the scores from those parts are not included in the final confidence. In the following treatment, we present a method to divide Latent SVM bias into component parts, which are then used to create CoP detectors.

The bias b in Eq. 4.1 in Latent SVM [24] is optimized such that confidences of positive examples are greater than 0, while those of negative examples are less than 0. This means that the bias balances the sum of filter and deformation scores from different parts among the positive and negative examples. Therefore, we divide the bias into constituent parts by averaging the contribution of each part to the positive and negative examples while ensuring that the sum of part biases sums to the Latent SVM bias b. Let j and k index positive and negative training examples, respectively. Then, the sum of confidences from the N^+ positive examples using Eq. 4.1 is given by:

$$\mathcal{S}^+ = \sum_{j=1}^{N^+}\left(\sum_{i=0}^{n} F_i'.\phi(H^j, p_i^j) - \sum_{i=1}^{n} d_i.\phi_d(d_{x_i^j}, d_{y_i^j}) + b\right). \tag{4.8}$$

Similarly, the sum of confidences from all negative examples is given by:

$$\mathcal{S}^- = \sum_{k=1}^{N^-}\left(\sum_{i=0}^{n} F_i'.\phi(H^k, p_i^k) - \sum_{i=1}^{n} d_i.\phi_d(d_{x_i^k}, d_{y_i^k}) + b\right). \tag{4.9}$$

Isolating the bias by multiplying Eq. 4.8 with $\mathcal{S}^-$, Eq. 4.9 with $\mathcal{S}^+$, and subtracting former from the latter:

$$\begin{aligned}
&(\mathcal{S}^- N^+ - \mathcal{S}^+ N^-)b \\
&\quad = \mathcal{S}^+ \sum_{k=1}^{N^-}\left(\sum_{i=0}^{n} F_i'.\phi(H^k, p_i^k) - \sum_{i=1}^{n} d_i.\phi_d(d_{x_i^k}, d_{y_i^k})\right) \\
&\quad - \mathcal{S}^- \sum_{j=1}^{N^+}\left(\sum_{i=0}^{n} F_i'.\phi(H^j, p_i^j) - \sum_{i=1}^{n} d_i.\phi_d(d_{x_i^j}, d_{y_i^j})\right).
\end{aligned} \tag{4.10}$$

Now, we simply decompose the bias into parts b_i under the assumption $b = \sum_{i=0}^{n} b_i$. Define $\varrho = \mathcal{S}^- N^+ - \mathcal{S}^+ N^-$. For deformable parts $i \in 1, ..., n$ we have,

$$\begin{aligned}
b_i = {} & \frac{\mathcal{S}^+}{\varrho} \sum_{k=1}^{N^-}\left(F_i'.\phi(H^k, p_i^k) - d_i.\phi_d(d_{x_i^k}, d_{y_i^k})\right) \\
& - \frac{\mathcal{S}^-}{\varrho} \sum_{j=1}^{N^+}\left(F_i'.\phi(H^j, p_i^j) - d_i.\phi_d(d_{x_i^j}, d_{y_i^j})\right),
\end{aligned} \tag{4.11}$$

and for root filter $i = 0$,:

$$b_0 = \frac{\mathcal{S}^+}{\varrho} \sum_{k=1}^{N^-}\left(F_i'.\phi(H^k, p_i^k)\right) - \frac{\mathcal{S}^-}{\varrho} \sum_{j=1}^{N^+}\left(F_i'.\phi(H^j, p_i^j)\right). \tag{4.12}$$

The bias for CoP detector $\mathbb{C}$ is the sum of bias of its constituent parts, given by: $b_{\mathbb{C}} = \sum_{\{i|p_i \in \mathbb{C}\}} b_i$.

The above procedure allows us to detect combinations of different body parts without requiring annotations for them. This advantageous outcome is due to Latent SVM as it infers the location of body parts using training examples when only provided with full-body annotations. Furthermore, although we have found the equivalence between different combinations at zero threshold, $\delta = 0$, the CoP detectors have different sensitivities to changes in δ. For that, we find the linear relationship between confidences of CoP detectors and the full-body detector using the confidences obtained on N^+ positive examples.

Another issue relevant to multi-object detection is non-maximal suppression (NMS). For dense crowds, heads are almost always visible since the only way to capture such a scene is by having the camera at some height. For NMS, we used heads as anchors of detections, and performed NMS only on bounding boxes of heads. We used the same procedure for generating all results, including baseline, to ensure consistency in post-processing.

4.4 Global Occlusion Reasoning (GOR)

Using the approach described in previous section, we obtain a dense set of detections, along with the scores of individual parts from CoP detectors. The resulting detections have significant overlap in crowded scenes, due to high density of individuals. On the other hand, it is possible that a completely visible human has a smaller bounding box, due to a relatively higher confidence generated by a CoP detector with fewer parts. The goal of occlusion reasoning is to expand and contract the bounding boxes so that they only but entirely cover the visible parts. And due to cyclic dependencies among individuals (A occluding B, B occluding C, ...) present in crowds, we propose to use efficient solutions using Binary Integer Programming (BIP) to solve this problem.

The formulation of Binary Integer Programming is given by:

$$\begin{aligned} \underset{z}{\text{argmin}} \quad & \mathbf{f}^T z \\ \text{s.t.} \quad & \mathbf{A}z \le \mathbf{b}, \\ & \mathbf{C}z = \mathbf{d}, \\ & z \;\text{ is a binary vector.} \end{aligned} \tag{4.13}$$

where **f** contains preferences associated with selecting the corresponding variables in z, which for our problem index over parts from all detections in an image. Independent of the output of CoP detectors, all parts go into the minimization. The scores of parts contributing to CoP detection are increased by confidence of that detection, while rest of the parts are given the raw scores obtained using full-body detector. Taking the negative of these values gives us the desired **f**. However, without the constraints, entire humans, whether occluded or not, will be selected as output. To get non-trivial solutions, we introduce several linear constraints. The first one is based on overlap i.e. if two parts from different individuals have significant overlap, then only one of them should be selected. A single constraint is of the form:

$$\begin{bmatrix} 0 & \dots & \mathbf{1}_{o(i,j)>\omega} & \dots & 0 & \mathbf{1}_{o(j,i)>\omega} & \dots & 0 \end{bmatrix} z \le \begin{bmatrix} 1 \end{bmatrix} \tag{4.14}$$

where $o(i, j)$ is the overlap of part i with part j from two different detections, which we obtained by dividing the overlap between parts i and j with the total area of i and j. Thus, for our case $o(i, j) = o(j, i)$. The indicator function outputs 1 if overlap is greater than ω. The constraint states that the overlap between two parts which are selected should be less than the ω, if it exceeds ω, one of the parts should be set to invisible. Each of these constraints forms one row of the matrix **A** and vector **b** in Eq. 4.13.

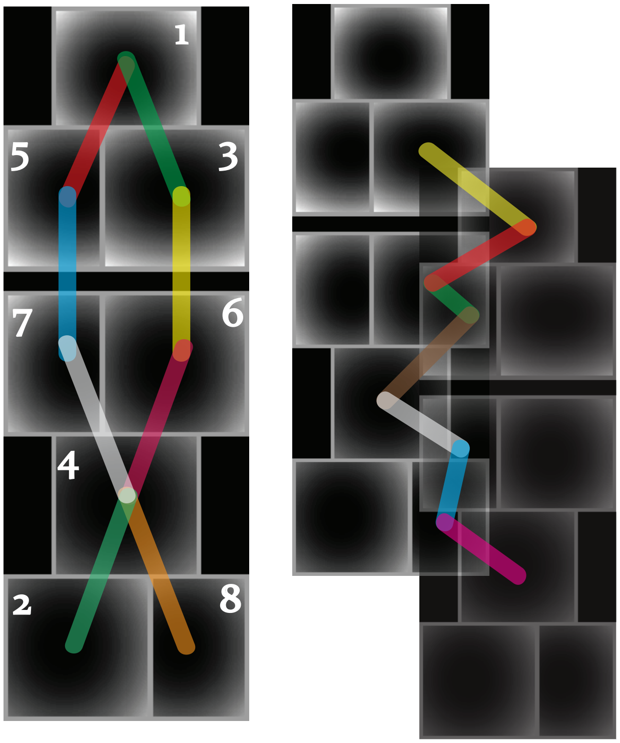


Figure 4.6: Linear constraints for Binary Integer Programming: Left shows the DPM model for a single person and the respective part numbers. To ensure all parts selected by IP are contiguous, we use chain constraints between parts as shown with different colors. Similarly, models for two occluding persons are shown on right. The overlap constraints ensure that occluded parts are rejected by the algorithm, thus giving bounding boxes consisting of visible parts only.

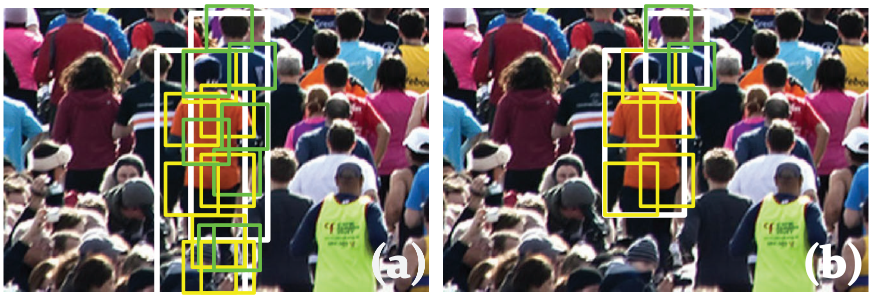


Figure 4.7: Results of Occlusion Reasoning: Two individuals are shown in (a) with their bounding boxes for root and deformable parts. (b) After reasoning for occlusion, only visible parts are selected, thus resulting in better localization.

Overlap constraints alone may result in degenerate solutions, where parts from the head and legs are selected by the optimization while those belonging to shoulders or abdomen are deselected. To alleviate this, we introduce chain constraints which ensure that only contiguous parts of all individuals are selected when dealing with overlap constraints. For a single detection Ω_q, let its corresponding part visibility variables be given by z_q. Using the part numbers given in Figure 4.6, the chain constraints are given by $Bz_q = \mathbf{0}_{7\times1}$ where,

$$B = \begin{bmatrix} -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & -1 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.15}$$

Thus, we have one such set of constraints per detection where each matrix essentially enforces the condition that the part below in a detection should only be selected, if the part above is also selected. These constraints are generated automatically by traversing the human model from top to bottom. The relationship between parts 4, 6 and 7 is different from the rest, that is the part 4 can be selected if either part 6 or 7 is selected. For all the detections, chain constraints are written as $\mathsf{diag}(B)z = \mathbf{0}_{Qn\times1}$ ($\mathbf{C}z = \mathbf{d}$ in Eq. 4.13) where Q is the number of detections, n is the number of parts per detection, and diag operator constructs a block-diagonal matrix using the argument with all other entries set to zero. Finally, the last set of constraints ensure that the output of CoP detectors are not violated. We allow inclusion of new parts immediately below those selected by a CoP detector, for instance, head and shoulders detection $\mathbb{C}_s$ is allowed to become upper-body $\mathbb{C}_u$, whereas the parts further below are hardwired to zero. Although the problem is NP-hard, exact inference is possible for our problem size, as z has $Qn \times 1$ dimensions, and overlap constraints only occur between neighboring detections. We used IBM CPLEX to solve the BIP problem. Figure 4.7 shows the results where initial bounding boxes of human and parts are shown in Figure 4.7(a) while results of occlusion reasoning are shown in Figure 4.7(b).

4.5 Experiments

We performed experiments on a challenging set of 108 crowd images, downloaded from Flickr. The images cover a variety of scenes and crowd densities, as some are sparse while other are dense. Some of the images depict marathons containing humans in standing poses, while other images are of parks and offer more difficult poses. Similarly, severity of occlusion also varies, as in some images, full body detection is possible, while in others, only heads are visible. We manually annotated the images for both heads and visible parts of humans. In total, there are ~$35,000$ bounding boxes for head and human each, making UCF-HDDC one of the largest and challenging dataset for Human Detection in Dense Crowds. There are two reasons for annotating heads separately from humans: 1) head bounding boxes can be converted to dotted annotations which mark presence of a human, and thus makes the annotations useful for counting as well. 2) In dense crowds, heads offer a much better estimate of detection accuracy as evaluation on human bounding boxes quantifies the quality of bounding boxes produced as well. The actual number of human annotations for each image are shown in Fig. 4.8. This dataset differs from the counting and localization dataset introduced in previous chapter which only has dotted annotations instead of bounding boxes and average number of people is an order of magnitude higher than this dataset.

We trained the human detector on INRIA person dataset [25], and used it directly on the proposed dataset. The biases $b_{\mathbb{C}}$ for CoP detectors were also computed on INRIA. For experiments, we used 100 images for testing and 8 for validation. The parameters of the model and hyper-parameters were set on the validation set. The values are $\eta = 0.1$, $\tau = 300$, $\sigma = 300$ $\alpha = 0.4$,

$\beta = 225$ and $\omega = 0.1$. We used the same value of ω for quantitative evaluation as well, i.e. using 10% overlap. The method is robust to changes in MRF and Influence Function parameters, with 50% change resulting in 1% drop in precision at 40% recall. However, 20% change in α and β results in almost 5% percent drop in precision.

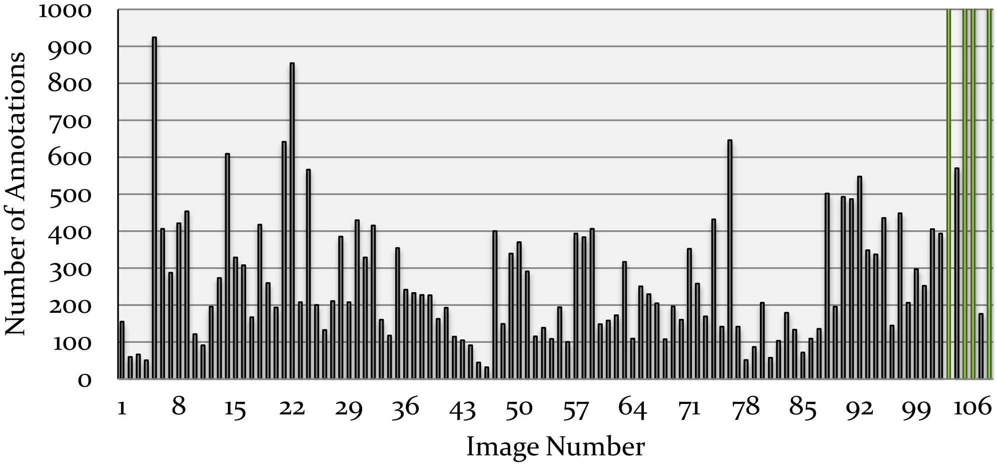


Figure 4.8: Statistics on the proposed UCF-HDDC dataset: On x-axis is the image id, while y-axis shows the number of human annotations in the image. The four green bars at the end have counts of 1276, 1852, 2816 and 2845, respectively.

4.5.1 Qualitative Results

Before we present quantitative results, we visualize the improvements obtained by the three ideas presented in this chapter in Figure 4.9. In this figure, green bounding boxes represent false negatives, black boxes show false positives, while the colored (red to yellow) bounding boxes represent true positives, where brighter colors signify greater overlap with ground truth annotations. The first row shows the gain in performance obtained using the proposed CoP detectors, shown in Fig. 4.9(b), over the full-body human detector, shown in Fig. 4.9(a). Results in both images are shown at 80% precision, thus, a higher recall means better performance. The additional humans that were detected are highlighted with yellow arrows.

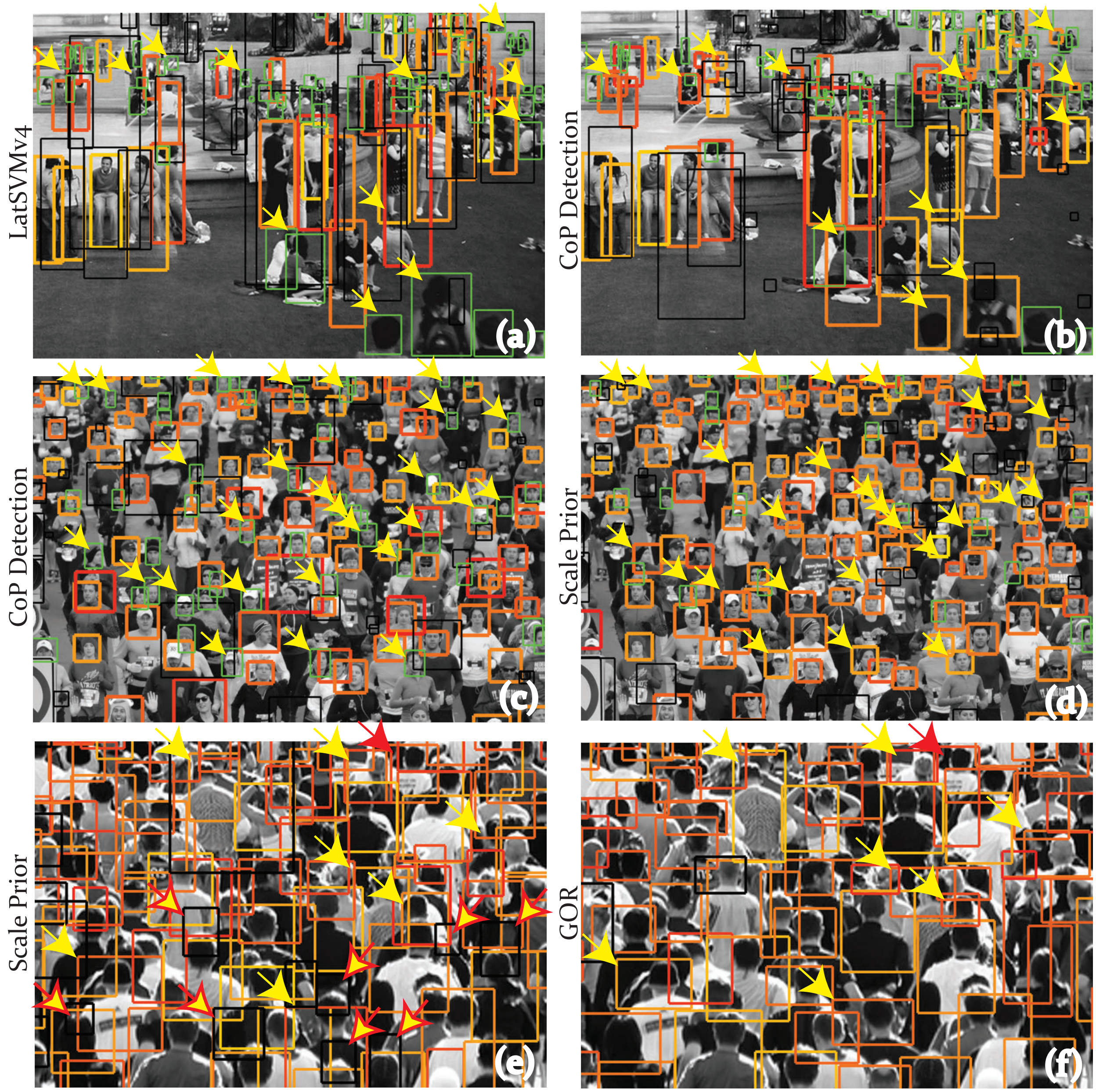


Figure 4.9: This figure visualizes the improvements using the three aspects of the proposed approach. Green boxes show false alarms, black represent false positives, while colors in the red to yellow range represent correct detections. In the first row, we show the improvement obtained by using combinations-of-parts detection. Results in both images are shown at the same precision, thus, a higher recall means better performance (shown with yellow arrows). The second row shows gain in performance obtained by using priors in addition to CoP detection. For clarity, only bounding boxes for heads are drawn. Similarly, the last row shows the results of Global Occlusion Reasoning (GOR). Yellow arrows indicate improved boxes, yellow-in-red arrows highlight false positives that were removed, while red arrow shows a box that worsened after GOR.

The second row in Figure 4.9 shows the improvement by using scale and confidence priors in addition to CoP detection. The bounding boxes corresponding to heads are shown for clearer visualization. These images also depict results at 80% precision, new detections being highlighted by yellow arrows. The third row presents some results of improvement using Global Occlusion Reasoning in addition to CoP detectors and priors. The bounding boxes in the Fig. 4.9(f) have much less overlap with each other than those in Fig. 4.9(e). Again, the yellow arrows highlight the improvements - the locations where the occlusion reasoning improved the quality of bounding boxes, the yellow-in-red arrows show the false positives which were removed as a result of reasoning, while red arrows shows a failure case where bounding box became worse. Still, the improvements outnumber the deteriorations, and thus leading to an overall increase in accuracy as suggested by quantitative analysis presented in the next subsection. Final results on three complete images are shown in Fig. 4.10. In this figure, white bounding boxes signify true detections (TP), black boxes indicate false alarms (FP), while green represents miss-detections (FN). In 4.10 (a), the crowd is sparse with humans inclined at an angle due to camera position. In 4.10 (b), the humans appear in varied poses, whereas 4.10 (c) is characterized by severe occlusions. The proposed approach gives excellent results for all three scenarios.

4.5.2 Quantitative Results

Figure 4.11 shows quantitative results of the proposed method evaluated with human bounding boxes using Precision vs. Recall, Miss Rate vs. FPPI (false positives per image), and Multiple Object Detection Precision or MODP. For the first two, we used an overlap of 10%, whereas MODP has overlap threshold as the x-axis obtained at 35% recall. The first two graphs show that on average, each module of proposed approach improves the performance, with scale and confidence priors and CoP detectors being equally crucial to increase in performance. On the other hand, MODP measures the quality of bounding boxes irrespective of false positives and negatives. The improvement from LatSVMv4 to CoP detectors in terms of MODP is obvious as it is able to pick up occluded humans while incurring fewer false positives. The improvement from CoP detectors to priors is due to change in proportion of true positives to false positives. Since priors reduce the hypotheses space, it reduces the relative number of false positives, which typically occur at random scales and may overlap with annotations. Similarly, since we used exactly the same bounding boxes for occlusion reasoning as were made available after priors, improved MODP suggests that occlusion reasoning results in better localization of detections.

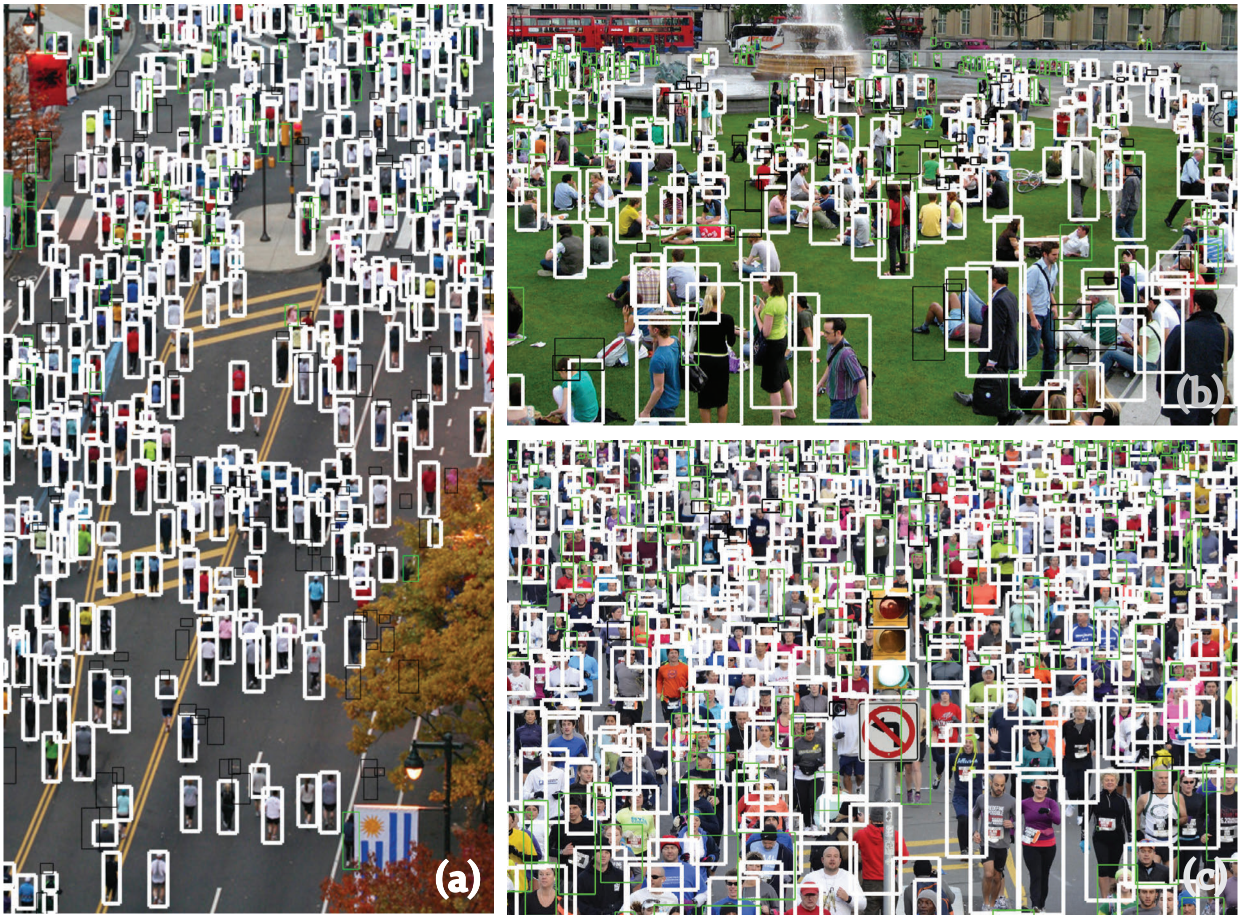


Figure 4.10: In this figure, white bounding boxes signify true detections (TP), black boxes indicate false alarms (FP), while green represents miss-detections (FN). In (a), the crowd is sparse with humans inclined at an angle due to camera position. In (b), the humans appear in varied poses, whereas (c) is characterized by severe occlusions.

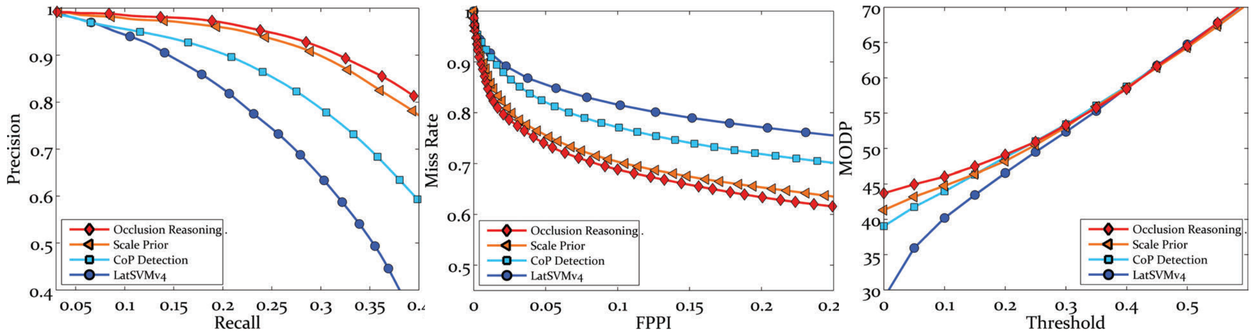
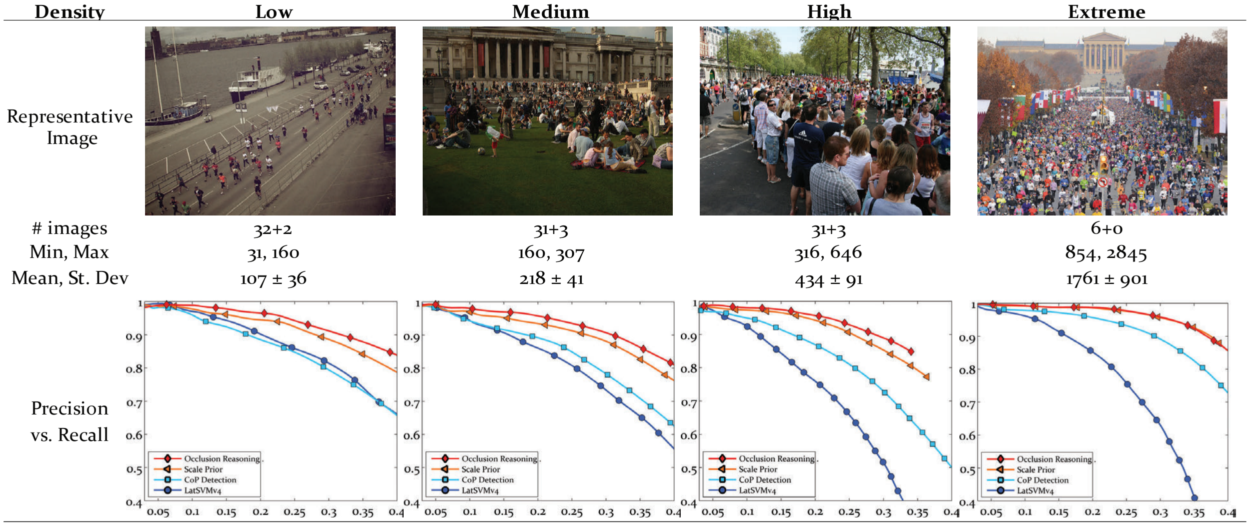


Figure 4.11: These graphs show the quantitative results highlighting the contributions of the three different aspects of the proposed approach. Curves in violet-blue show the results of LatSVMv4 [24] (baseline), blue represents CoP detection, orange depicts improvements from priors, whereas red highlights the improvements from Global Occlusion Reasoning.

4.5.3 Density-based Analysis

To test the robustness and contributions of the three aspects of our method with respect to size of crowd, we performed a density-based analysis in Figure 4.12. Here, we simplify the notion of density which refers to the number of humans per image rather than number of people per unit area in real world which is difficult to ascertain. Thus, we sorted the images according to number of annotations, and divided them into four groups: low, medium, high and extreme. In Figure 4.12, the first row shows representative images with median counts for each density group. The number of images and some statistics on the number of humans in each group is presented below the median images. Finally, the precision-recall curves are shown at the bottom of the figure. The curves offer important observations with respect to the three modules. The performance of CoP detection improves with increasing density, simply because humans in high-density undergo more occlusions. The scale and confidence priors give consistent improvement upon CoP detectors across all densities, which is around (15%). This means that scale and context is important at all densities. However, occlusion reasoning does not improve at extreme densities, which may be due to the bias of CoP detectors towards combinations with fewer parts for this density. Occlusion reasoning for this group only results in tighter boxes, not affecting the overlap with the predominantly small boxes in annotation, and thus, is not likely to show a noticeable improvement in precision.



Density	Low	Medium	High	Extreme
# images	32+2	31+3	31+3	6+0
Min, Max	31, 160	160, 307	316, 646	854, 2845
Mean, St. Dev	107 ± 36	218 ± 41	434 ± 91	1761 ± 901

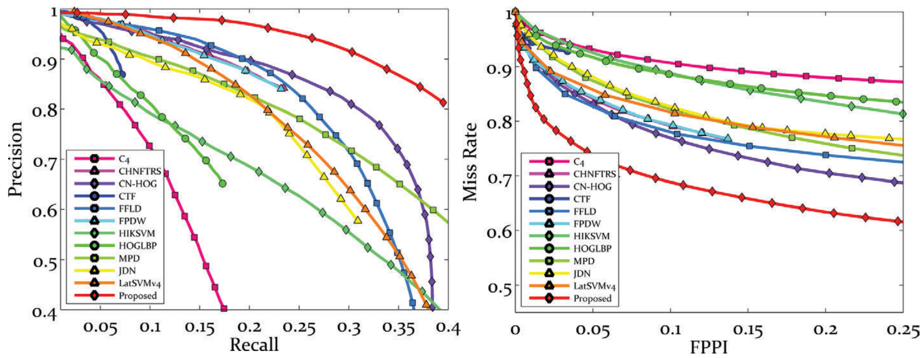
Figure 4.12: Density-based analysis: The evaluation on four different densities - low, medium, high and extreme. This figure shows the median image from each density, some statistics on the number of humans, followed by precision-recall curves for LatSVMv4 (baseline), CoP detection, scale and confidence priors and global occlusion reasoning. The addition in # images differentiates images in test and validation set.

4.5.4 Comparison

We compared the output of the proposed method to several other human detectors. We used the available pre-trained codes provided by the authors. Many work reasonably well in low to high density, but their performance deteriorates on extremely dense images due to severe occlusion. The comparison is shown in Figure 4.13 which also shows the Average Precisions along with abbreviated titles.

We used LatSVMv4 [24] as our underlying detection module trained on INRIA person dataset. There are several methods which outperform [24] on this dataset, but still, the proposed approach is able to perform better than all of the other methods. From Figure 4.13, we see that at 35% recall, the difference between the precisions of proposed and state-of-the-art methods is almost 15%. We believe using CN-HOG [54], which is also based on LatSVM, as underlying detection module will further improve the performance of our approach.

Furthermore, it is important to realize that the recall of proposed method is upper-bounded by that of CoP detection, which in turn is dependent on the underlying human detector (LatSVMv4 in our case). It is simply not possible to obtain more detections through the priors or occlusion reasoning than the underlying detection mechanism employed. For this dataset, recall curve hits the asymptote at around 50% which is low. Although this is due to the challenging nature of this dataset, we believe in order to obtain better performance for human detection in dense crowds, future research must be directed at improving CoP detectors.



Method	AP
C4	13.08
CHNFTRS	21.86
CN-HOG	33.62
CTF	6.92
FFLD	31.47
FPDW	21.6
HIKSVM	26.69
HOGLBP	14.42
MPD	32.1
JDN	25.71
LatSVMv4	31.04
Proposed	**37.2**

Figure 4.13: Comparison with C4[58], CHNFTRS[53], CN-HOG[54], CTF[56], FFLD[57], FPDW[27], HIKSVM[55], HOGLBP[26], MPD[98], JDN[99] in addition to LatSVMv4[24]. The proposed method outperforms all methods on both measures despite using an underlying detector [24] with lower performance than comparison methods.

4.5.5 Failure Cases

Due to crowded and challenging nature of the dataset, there are several failure cases. First, we highlight two cases where human or CoP detection is difficult. The first is related to low resolution. Figure 4.14(a) shows a small patch which is $1/400$-th the size of original image. Humans in this region become extremely blurred resulting in weak edges and deteriorated HOG-based detection, however the patches are still annotatable by humans. The human size also becomes small, causing issues for DPM as it has a lower limit on detectable part size at 23x23 pixels. The camera position relative to human height is also important, for instance, in Figure 4.14(b), even the heads are partially occluded resulting in poor initial detections by the CoP detectors. The solution is to have detectors that are robust to even partial occlusions of parts.

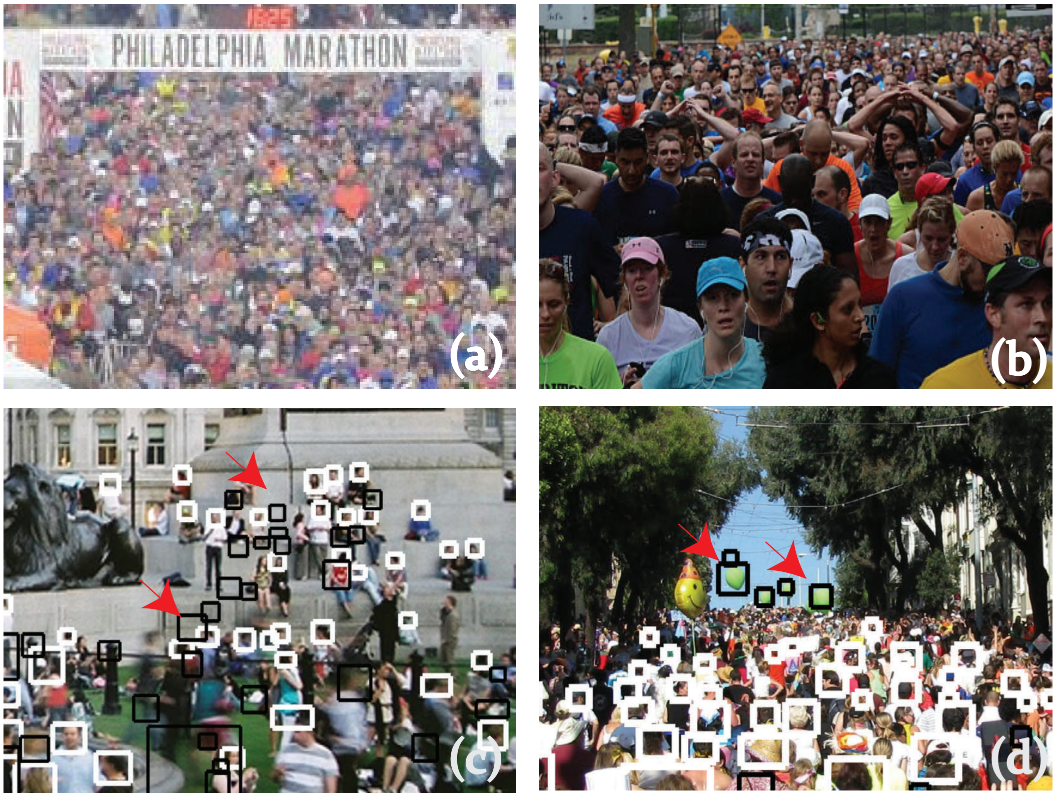


Figure 4.14: Failure Cases: (a) Blurring due to large distance from the camera in addition to small size due to perspective effects. (b) Camera position relative to humans in the scene may result in large number of occlusions where even the heads are partially occluded. (c) Hypersensitivity introduced due to priors caused by large number of high confidence detections in a low density region. (d) High-confidence false positives at the boundary of crowd may results in incorrect initialization of priors. In (c) and (d), true positives are shown with white, while false positives are shown in black.

Figures 4.14(c,d) show failure cases specific to scale and confidence priors. When we have high-confidence detections in first iteration of prior discovery in a region that has fewer humans per unit area, it sometimes makes the method hypersensitive to detection hypotheses occurring at the desired scale in neighboring areas. This is shown with red arrows in Figure 4.14(c). Similarly, high confidence non-human detections at early iterations also degrade the scale prior by providing incorrect scale information, thereby resulting in more miss-detections in their surroundings, as can be seen with the balloons in Figure 4.14(d).

4.6 Chapter Summary

In this chapter, we showed that context, employed in the form of locally-consistent scale and the associated confidence priors, is helpful in improving human detection in dense crowds. Using a novel function which measures the influence of each detection in its neighborhood, we build scale and confidence priors, which are then iteratively improved. And to detect partially visible humans, we proposed combinations-of-parts detection using different configurations of parts of a complete human by dividing the Latent SVM bias into constituent parts. Furthermore, we presented an Integer Programming formulation to the task of occlusion reasoning where we attempted to minimize overlap between detections and maximize contiguity of parts within each detection with preferences of selection determined by raw scores output by underlying combinations-of-parts detector. We evaluated our approach using a new set of difficult images, and showed that each aspect is important for detecting humans in dense crowds.

CHAPTER 5: TRACKING WITH PROMINENCE AND NEIGHBORHOOD MOTION CONCURRENCE

In this chapter, we present our approach for tracking individuals in dense crowds. First, we discuss the notion of prominent individuals and present an algorithm that identifies such individuals. Then, we present the Neighborhood Motion Concurrence model which gives a probability surface of position for an individual using position and velocity information of the target and its neighbors. Finally, we develop a tracking method which updates the position of individuals in an ordered fashion using information about prominent individuals, influence from neighbors, and feedback from template-based tracker.

5.1 Prominence

Although it is possible to track and update the positions of all individuals in a crowd simultaneously at each time step, this is not the most efficient method. Some individuals have unique characteristics that make them stand out from the crowd. These characteristics make it easier to establish correspondences across frames for these individuals without confusing them with the rest of the crowd. The first step, therefore, would be to detect prominent individuals whom we will refer to as *Queen Bees* or, in short, queens. We choose to use this term because a queen, due to its size, is the only unique bee in an entire colony of indistinguishable bees. Since a queen is unique and easily identifiable, it can be used to describe prominent targets in any type of dense crowd.

There are several features that can make an individual prominent with respect to others, such as gait [39], physical appearance, height or age. However, in dense crowds with a small number of pixels on a target, visual appearance is a robust and, typically, the only observable feature. In our framework, a queen is defined as individual with color features that differ significantly from the majority of the crowd.

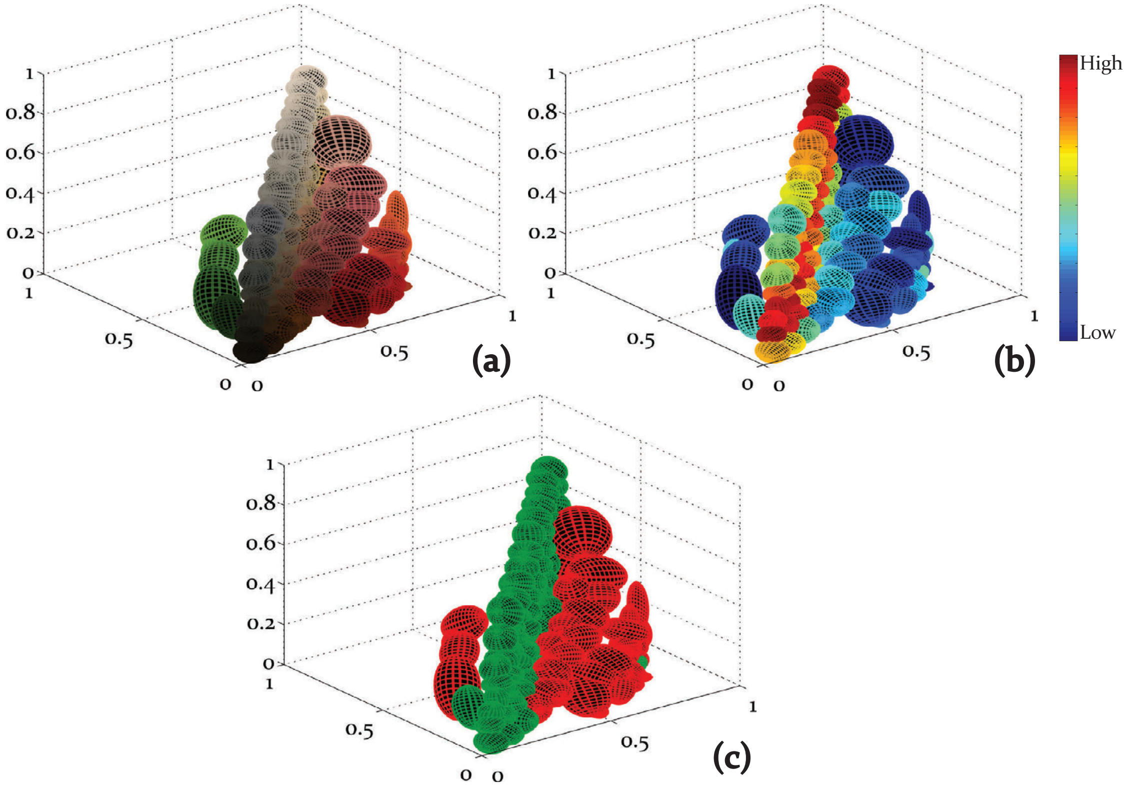


Figure 5.1: Visualization of clusters: Given a fixed set of templates, we extract $[R, G, B]$ features for each pixel in the template. We keep a map (Ω) between features and templates, i.e., we associate the id of each template with its constituent features. Then, the features are clustered and modeled using Mixture-of-Gaussians distribution. The results on the image and templates shown in Figure 1.4 are shown in (a) where each Gaussian is represented with an ellipsoid drawn with size equal to 1.5 the size of variance, i.e., $1.5 \cdot (\Sigma)^{1/2}$ and colored with its mean, i.e., μ. The colors belonging to non-queens (white templates in Figure 1.4) form clusters along the diagonal (black to white). In (b) we color the ellipsoid according to the density of the respective Gaussians with sparse clusters in blue and dense clusters in red. (c) The clusters that were used in selecting the queens are given in red after which the process of back-assignment stopped and did not use clusters drawn in green.

To select the queens, we extract features ϕ_i from the templates $T_{1:n}$. Every pixel in each template gives one 3d feature, i.e., $[R, G, B]$ at that pixel. While generating the features, we keep a map, Ω, that associates features to the source templates. All the features are then clustered into k clusters modeled using mixture-of-Gaussians distribution, i.e., each component is $\mathcal{N}(\mu, \Sigma)$. Next, the clusters $C^{1:k}$ are sorted in ascending order according to a particular criterion (density). Finally, the features are reassigned to their original or source templates beginning with features from first cluster in the sorted list. The process is stopped once a small percentage of total templates (in our case, 10%) are filled by at least two-thirds. Since all the features from each cluster are processed simultaneously, it is possible to have more than ten percent of total templates selected as queens. Algorithm 1 gives a general and formal description of this procedure.

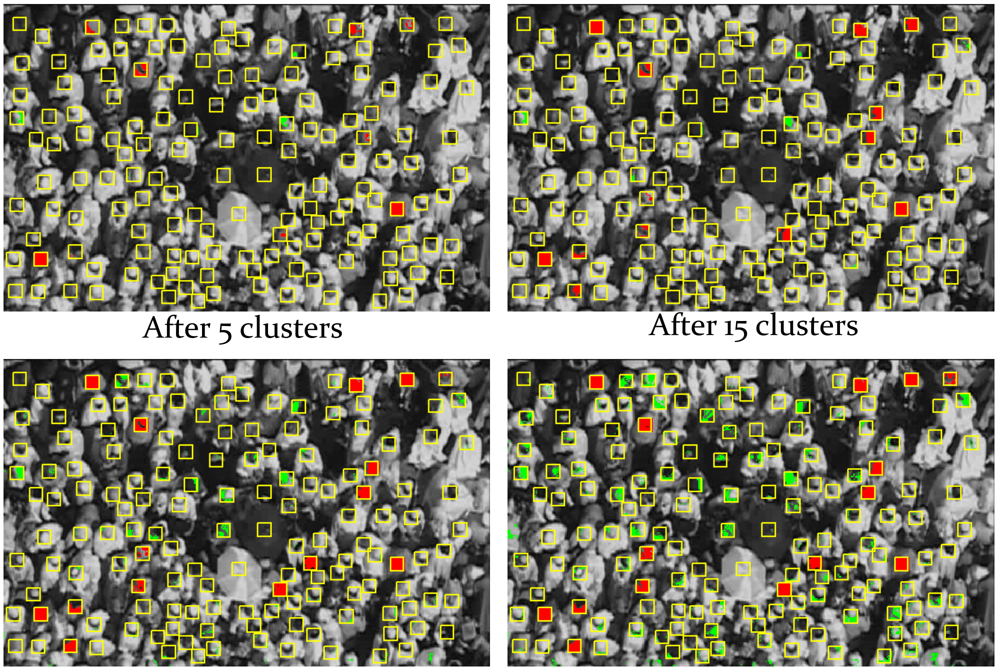
For the image given in Figure 1.4, the results of clustering are shown in Figure 5.1(a). In this figure, each cluster is drawn in the RGB space using an ellipsoid whose size and orientation is determined by Σ and the color is given by μ. In Figure 5.1(b), we color-code the clusters according to density where blue indicates sparse clusters and red indicates dense clusters. Figure 5.1(c) shows the clusters whose features were utilized during back-assignment. These clusters are shown in red and the features in these clusters primarily belong to queens, whereas the features belonging to clusters in green were not used because the desired number of queens had already been identified by back-assignment by that time. The intermediate results of back-assignment for the image in Figure 1.4 are shown in Figure 5.2 where the procedure stopped after processing 35 clusters. The final results are shown in Figure 1.4 where the yellow templates mark the selected queens while white templates belong to non-queens.

For a cluster C, its mass m can be given by $|C|$ (i.e. the number of data points in C), and volume v given by $(2\pi)^{3/2}|\Sigma|^{1/2}$. Then there are several ways to sort clusters: mass (m), volume (v), mass weighted by volume ($m \cdot v$), density (m/v) or the reciprocal of density (v/m). We ran a small experiment to find which of the criteria gives the best results for prominence by determining if the criterion correctly identifies the queens while filling few non-queen templates (red and green in Figure 5.2, respectively). We ran the experiment several times to avoid differences due to clustering, and found that density gives the correct and most stable queens across iterations. This implies that features of queens behave as outliers during clustering and form sparse clusters, whereas features of non-queens form dense clusters since they tend to be similar to each other.

Algorithm 1 Algorithm to find queens given templates $T_{1:n}$

```
procedure DETECTQUEENS
    $\Phi = [\ ]$                                          ▷ feature matrix
    for all $i = 1$ to $n$ do
        $\phi_i \in \mathbb{R}^{3 \times N_i}$ = features from $T_i$          ▷ feature =: an RGB vector per pixel
        Concatenate $\phi_i$ to $\Phi$
        $\Omega(\phi_i) = i$                               ▷ map from features to templates
    end for
    $[C, \mu, \Sigma]$ = GMM($\Phi, k$)                    ▷ $k$=# of clusters
    Sort clusters w.r.t density i.e. $|C|/(2\pi)^{3/2}|\Sigma|^{1/2}$

    $V_{1:n}$ =[0 0 ... 0]                                 ▷ 1xn voting array
    $queens = [\ ], i = 0$
    while $i \leq k \wedge |queens| < .1n$ do
        $i = i + 1$
        for all $\phi \in C^i$ do                          ▷ $C^i$ =: $i$th cluster
            $V_{\Omega(\phi)} = V_{\Omega(\phi)} + w(\phi, i)$       ▷ $w$=voting function, $w(\phi, i) = 1$ for our case
        end for
        $queens = \{j \mid V_j > \frac{2}{3} N_j\}$
    end while
end procedure
```



After 5 clusters

After 15 clusters

After 25 clusters

After 35 clusters

Figure 5.2: Intermediate outputs for the queen detection method: The images correspond to back-assignment after processing $k = 5, 15, 25$ and 35 clusters (out of $k = 100$). Red and green colors indicate queens and non-queens, respectively. Notice that the proportion of green regions to red increases as the number of clusters increases.

5.2 Neighborhood Motion Concurrence (NMC)

In this section, we present an intuitive model, that utilizes the dynamic contextual information of the crowded scene, which allows us to track individuals in a dense crowd without requiring any prior knowledge (crowd flows, motion patterns, ...). Let $x_i^t = [\mathrm{x}\ \dot{\mathrm{x}}]^T$ (position, velocity), Σ_i^t represent the state and covariance, respectively, of target i at time t, $\hat{x}_i^t$ be the updated state, A be the state transition matrix, for instance, linear velocity and $\mathcal{N}(\mu, \Sigma)$ a $2d$ Gaussian distribution. We will distinguish the target under consideration from its neighbors by using subscripts i and j, respectively.

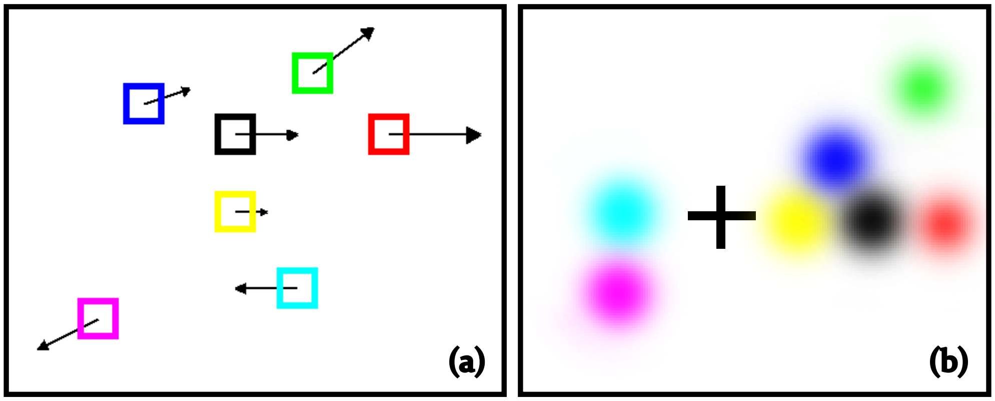


Figure 5.3: Visualization for Neighborhood Motion Concurrence (NMC) model: (a) The target under consideration whose position is to be updated is shown with black square while its updated neighbors are shown with different colors. The arrows show the velocity which, for the target is velocity at $t-1$, whereas, for neighbors is their velocity at t. (b) shows the probability of position using the model for the target in (a), given by Equation 5.5. The cross hair represents position of the target before the update, i.e., position at $t-1$. Each blurred circle represents $\mathcal{N}(\mu, \Sigma)$. The black circle is obtained using constant velocity assumption on the motion of target (p_S from Equation 5.1), while colored circles capture the influence from neighbors (p_N from Equation 5.2), based on their respective motions. This is a simple illustration, so each covariance is assumed to be an identity matrix.

Neighborhood Motion Concurrence has two components, namely self p_S, and neighbors' influence p_N. Since the state of the target under consideration at time t has not been updated yet, both its position and velocity come from the previous time instant $t-1$,

$$p_S = p(z_i^{t-1} \mid x_i^{t-1}) \cdot \mathcal{N}\left(Ax_i^{t-1}, A\Sigma_i^{t-1}A^T\right), \tag{5.1}$$

where $p(z_i^{t-1} \mid x_i^{t-1})$ denotes the observation likelihood z_i^{t-1} of the target given its state x_i^{t-1} obtained through underlying tracker, and A is the state transition matrix which captures linear velocity for our case. If there was some uncertainty in the target's position at time $t-1$, then p_S gets weighed down by this factor, therefore more preference will be given to prediction from neighbors, which is the second component of NMC, given by,

$$p_N = \sum_j w_j \cdot \mathcal{N}(Ax_{ij}^t, A\Sigma_j^t A^T) \cdot \lambda_j, \tag{5.2}$$

where $x_{ij}^t = [\mathrm{x}_i^{t-1} \; \dot{\mathrm{x}}_j^t]$, $\lambda_j = 1$ if the state of target j has been updated before i at time t, and 0 otherwise. But, not all influences can be treated equally, so we weigh them according to the neighbors' distance from the target,

$$w_j = \frac{\exp(-\|\mathrm{x}_j - \mathrm{x}_i\|)}{\sum\limits_{k \in Neighbors} \exp(-\|\mathrm{x}_k - \mathrm{x}_i\|)}. \tag{5.3}$$

To illustrate the idea, consider the target shown in black square in Figure 5.3(a) whose position is to be updated at time t, i.e., the black square is drawn where the target was at time $t-1$. Its updated neighbors are shown with squares of different colors, whose positions are depicted at time t. The arrows originating from the center of squares indicate the velocity of each individual, i.e., velocity at time $t-1$ for target and at time t for the updated neighbors. In Figure 5.3(b), we show how each velocity vector from Figure 5.3(a) influences the likelihood of the target's position. In this image, the cross-hair marks the position of target before it is updated. The blurred circles represent Normal distributions ($\mathcal{N}(\mu, \Sigma)$). The black circle represents p_S, while colored circles represent p_N, using the same colors as the squares depicting neighbors in Figure 5.3(a). Here, all covariances are set to identity matrices for the sake of visualization. Thus, NMC generates a probability distribution which gives a dynamic prior on the motion of the target based on its own motion and that of its neighbors. Figure 5.4(a) shows a real example of the use of this model. The position of individual in white square with cross-hair is to be updated, while some of its neighbors have already been updated, shown in colored squares. The lines originating from the center of squares show the velocity vectors. Figure 5.4(c) is the output of the model, which is a multi-modal distribution with one strong peak. Figure 5.4(d) shows the probability of target's position using just the appearance, while Figure 5.4(e) shows the drastic reduction in confusion in the target's position achieved with the model, which is typical to majority of the non-queen individuals of the crowd. The final results are shown in Figure 5.4(b) where black star represents the incorrect position updated without using NMC, and red star indicates the correct position updated with NMC.

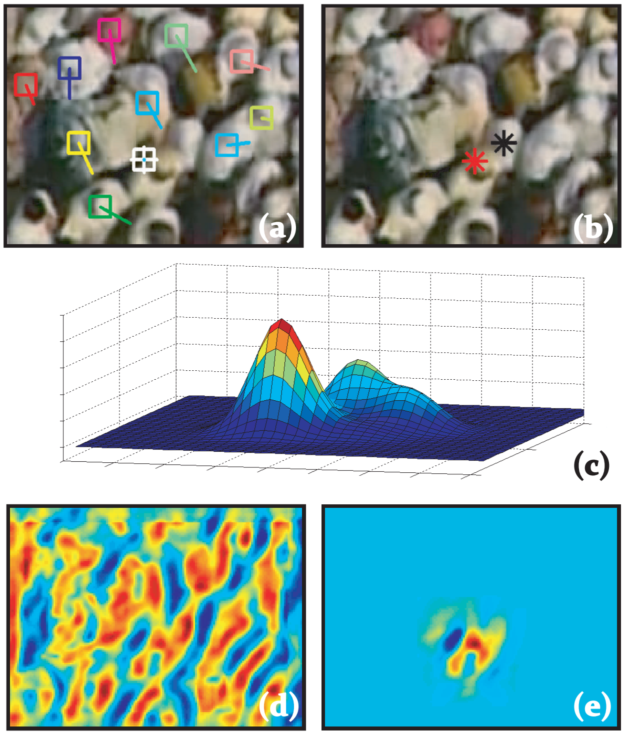


Figure 5.4: (a) The target under consideration is shown in white and its updated neighbors in color. (b) The red star is the correct position updated by the method, whereas black star is the incorrect position update from template-based tracker alone. Intermediate results: (c) is the probability surface of position using NMC for the target in (a). The bottom row shows the effects of using the model, where (d) and (e) are probability surfaces of position before and after the model is applied.

5.3 Appearance based Instantaneous Flow

Neighborhood Motion Concurrence models the similarity of motion of individuals in a dense crowd. The assumption that an individual has motion similar to its neighbors is violated when there are multiple flows in close vicinity of each other, for instance, two groups of people walking right next to each other in opposite directions, or when there are anomalous individuals in the crowd whose motion is not consistent with their neighbors. In these cases, we revert to Instantaneous Flow (Figure 5.5), which provides some information about the possible direction of motion for such individuals. We construct Instantaneous Flow from five frames using normalized cross correlation on patches that are densely initialized throughout the scene, where track of each patch captures temporally-localized motion. The idea is similar to particle advection [9], however when the duration is only five frames, particle advection gives results significantly worse than instantaneous flow, due to noisy and inconsistent optical flow. We initialize 4x4 patches at a regular spacing of 4 pixels. When NMC doesn't provide good prediction, e.g., when the observation likelihood at the updated position is low, we approximate the motion using nearby patches in instantaneous flow field. In this case, the neighborhood p_N component changes and $x_{ij}^t = [\mathrm{x}_i^{t-1}\ \dot{\mathrm{y}}_j^t]$, where $\dot{\mathrm{y}}_j^t$ is the velocity of the patch averaged over the five frames. The selection of neighbors in Equation 5.3 is now based on nearest patches instead of individuals.

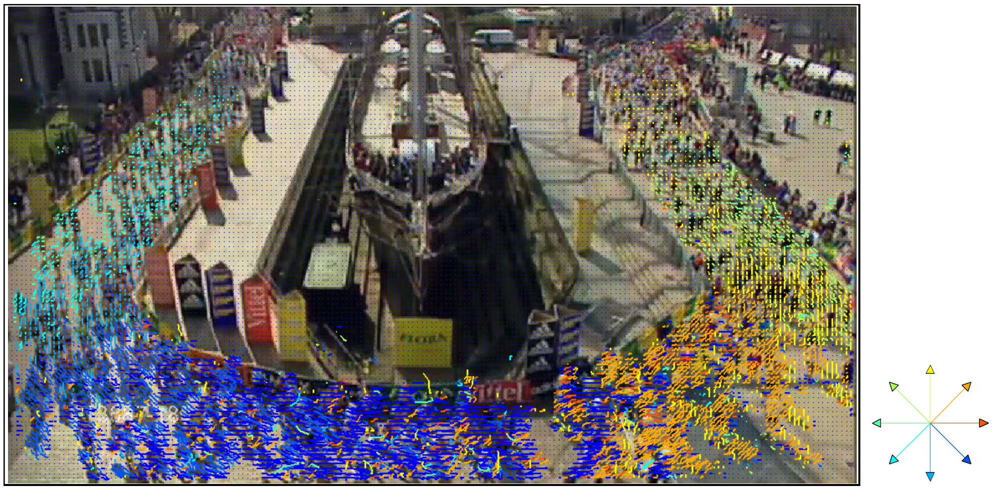


Figure 5.5: This figure shows instantaneous flow computed for one of the frames of Sequence 5. The patches were densely initialized at a spacing of 4 pixels. The direction at each location in the image is shown with color wheel on bottom-right of the image.

5.4 Tracking in Dense Crowds

In this section, we use the three aspects proposed in previous sections together in a joint framework, which allows us to formulate a solution to the challenging problem of tracking in dense crowds without relying on any prior knowledge.

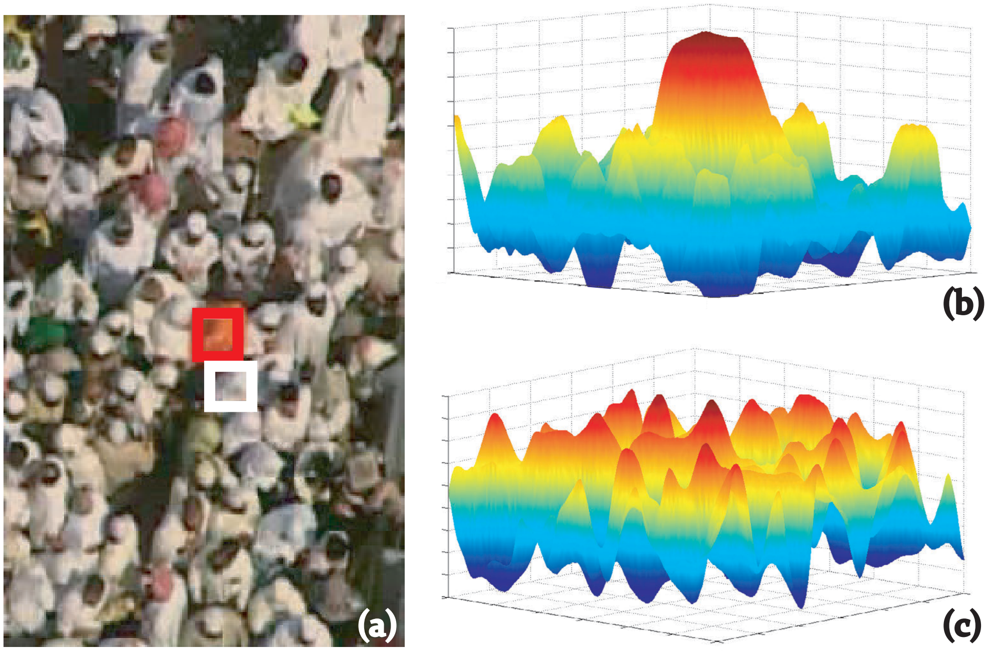


Figure 5.6: (a) The red square marks one of the queens and the white one is its immediate non-queen neighbor. (b) shows the probability surface (obtained through Equation 5.6) of the queen's position in the next frame while (c) is the corresponding probability surface for its neighbor. It is obvious that queens, due to uni-modal probability distribution, have less confusion in maintaining identity than non-queens, and therefore should be placed at the top of tracking hierarchy.

Given some initialization, our goal is to track each individual in the crowd. If the crowd flow has been modeled in advance, then it is possible to update the positions of all individuals simultaneously. However, a non-aggressive approach is preferable when the flow is not known. For each individual at each time instant, a decision needs to be made for the position update, which allows us to assign a confidence to this decision. Thus, tracking can be posited as a decision making process where queens serve as guides and NMC is used for consensus. This idea lends itself to a hierarchical framework which starts with queens and ends with non-queens.

At the top of the tracking hierarchy are queens, which are updated first. Figure 5.6 justifies their placement at the top of hierarchy. The two targets, one queen and an adjacent non-queen are shown with red and white squares, respectively. In Figure 5.6(b), we show the probability surface of position using just the appearance for the queen, while in Figure 5.6(c), we show the same for the non-queen. The surface in Figure 5.6(c) is common to non-queens which signifies greater possibility of confusion among them, in this case, due to white appearance of nearly all the neighbors. It is evident that the queen's neighbor in Figure 5.6(c) will pose a significant challenge in tracking unless its state predictions are guided by the adjacent queen.

Once the queens are updated, their immediate neighbors are updated next, then the neighbors of neighbors. This process continues to expand outward until every target has been updated at the current time. If $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denotes the graph where $\mathcal{V}$ is the targets represented by their states and $\mathcal{E}$ is the edges between each individual and its neighbors within a fixed radius, then the updating order is Breadth First Search which can be implemented using a queue.

Let $NN_i \equiv \{j \mid e_{ij} \in \mathcal{E}\}$ be the neighbors for target i. For simplicity of notation, index j will represent a member of NN_i. Given states and covariances of a target i and its neighbors j, under first-order Markov assumption,

$$\begin{aligned} &p\left(x_i^t \mid z_i^t, x_i^{t-1}, x_j^t\right) \\ &\quad \propto p\left(z_i^t \mid x_i^t\right) p\left(x_i^t \mid x_i^{t-1}, x_j^t\right) . \end{aligned} \tag{5.4}$$

The state of queens is updated first, which is predicted only using their previous state, defined as $p\left(x_i^t \mid x_i^{t-1}, x_j^t\right) = p_S$. However, the neighbors of a non-queen target whose state has been updated at time t influence its state estimate using NMC, approximated by the following function,

$$p\left(x_i^t \mid x_i^{t-1}, x_j^t\right) = \zeta\left(p_S + p_N\right), \tag{5.5}$$

where ζ is the normalization factor.

In Equation 5.4, $p(z^t \mid x^t)$ is the probability of measurement given state at time t, which corresponds to confidence from the tracker. Our underlying tracker is template-based, with Normalized cross-correlation as the similarity measure, hence,

$$p(z^t \mid x^t) = \frac{1}{2}\left(\gamma(\mathrm{x}^{\mathrm{t}}) + 1\right), \tag{5.6}$$

where $\mathrm{x} = (u, v)$ and $\gamma(\mathrm{x})$ is given by

$$\frac{\sum\limits_{x,y,z} \bar{f}(x,y,z) \cdot \bar{T}(u-x, v-y, w-z)}{\sqrt{(\sum\limits_{x,y,z} \bar{f}(x,y,z))^2 (\sum\limits_{x,y,z} \bar{T}(u-x, v-y, w-z))^2}}, \tag{5.7}$$

where z iterates over color channels and $\bar{T}$ is the target template T with its mean subtracted.

Finally, the state with maximum posterior probability is given by,

$$\hat{x}_i^t = \underset{x_i^t}{\operatorname{argmax}}\, p\left(x_i^t \mid z_i^t, x_i^{t-1}, x_j^t\right). \tag{5.8}$$

Algorithm 2 Algorithm to update states given templates: $T_{1:n}$, NN graph: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, state vectors at time $t-1$: $x_{1:n}^{t-1}$, and id of queens: $Q \subset \{1, 2, \ldots n\}$

```
procedure HIERARCHICALUPDATE
    $\forall i \mid i \in \{1, 2, \ldots, n\}, \lambda_i = 0, \eta_i = 0$        ▷ $\lambda$: if updated, $\eta$: # visits
    Insert $\{q \mid q \in Q\}$ into queue
    while do $\exists j \mid \lambda_j = 0$
        Retrieve element $i$ from start of queue
        Generate $p(x_i^t \mid x_i^{t-1}, x_j^t)$ from NMC, and $p(z_i^t \mid x_i^t)$ according to Equation 5.6
        Find $\hat{x}_{i,NMC}^t$ by Equation 5.8
        if $p(z_i^t \mid \hat{x}_{i,NMC}^t) < \tau$ then
            Generate $p(x_i^t \mid x_i^{t-1}, x_j^t)$ based on instantaneous flow
            Find $\hat{x}_{i,IF}^t$ by Equation 5.8
        end if
        if $p(z^t \mid \hat{x}_{i,NMC}^t)^{\frac{1}{\eta_i+1}} > \tau \vee p(z^t \mid \hat{x}_{i,IF}^t) > \tau$ then
            $\lambda_i = 1$
            Push $\{j \mid j \in NN_i \wedge \lambda_j = 0\}$ into queue
            in order of increasing distance
        else
            $\eta_i = \eta_i + 1$
            if $i \in Q$ then
                $Q = Q \backslash \{i\}$
            end if
            Push $i$ at the end of queue
        end if
    end while
end procedure
```

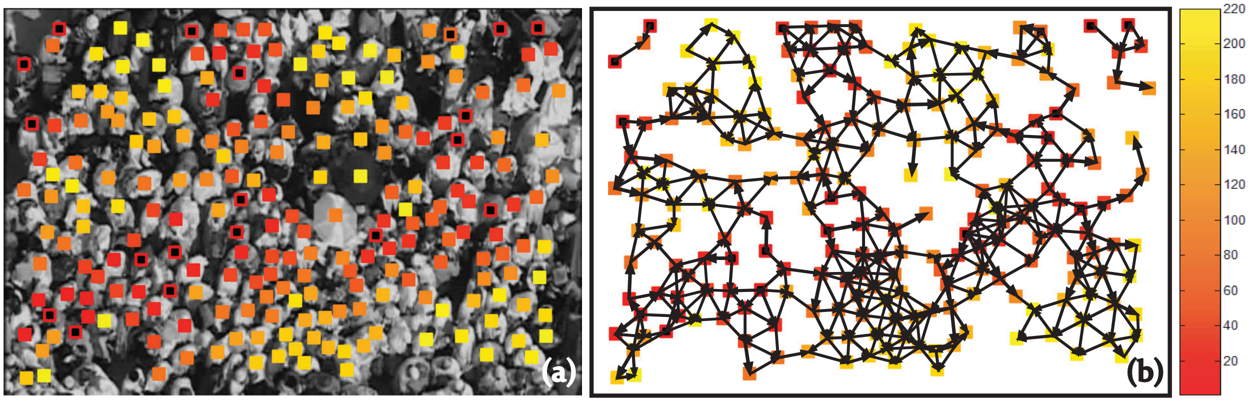


Figure 5.7: Hierarchical Update: (a) This image shows the order of update from Algorithm 2 on Sequence 2 which contained 220 people. The colors are encoded with the color-bar on the right, with red indicating individuals that were updated earlier than the ones shown in yellow. The update scheme starts with queens (black square inscribed in a red square), and moves down the *hierarchy* till all of the individuals are updated. Notice the occurrence of yellow squares in proximity with red and orange, which depicts delayed update. (b) shows the DAG produced as a result of the hierarchical update where edges, shown with arrows, signify the direction in which the influence was transmitted.

Note that the distribution in Equation 5.5 depends only on the neighbors whose position has been updated. If $p(z^t \mid \hat{x}^t)$ for a target is low, which might be due to poor prior probability from NMC or occlusion, we then resort to instantaneous flow for obtaining prior probability. The prior probability for such individuals is based on neighboring patches in the instantaneous flow field which is similar to Equation 5.2 except that x_{ij}^t now comes from patches rather than individuals. However, if the update confidence does not improve from using Instantaneous Flow, we delay the update of such individuals and place them at the end of the queue so that it doesn't influence rest of the crowd. Placing the target back into queue when tracker confidence is low has the peril of running into an infinite loop. The following theorem shows that it is not possible in case of Algorithm 2 which, in effect, gradually lowers the threshold till the individual's position is updated.

Theorem. *The number of times a target is visited (attempted for update) at time t is finite given $\tau < 1$.*

Proof. The target *revisited* can either be a queen or a non-queen. After the first visit, in case the queen fails condition in Algorithm 2, Line 12, the algorithm, at later visits, will treat it as non-queen with prior probability governed by Equation 5.5.

Let $p(z_i^{t,k} \mid \hat{x}_i^{t,k})$, denote the probability of observation on kth visit to a non-queen target i at time t. There can be two cases at kth visit:

Case 1. $\forall j \mid j \in NN_i, \lambda_j = 1.$

Case 2. $\exists j \mid j \in NN_i, \lambda_j = 0.$

For Case 1, the distribution from Equation 5.5 will not change for $l > k$. For Case 2, there are two possibilities: either for some $k' > k$, all the neighbors of the target get updated, which will collapse Case 2 to Case 1. The other possibility is when at least one of the neighbors is in the same situation as target i (i.e. $\lambda = 0$ for both). Under such circumstances, Equation 5.5 for $l > k$ visits will still not change, since λ_j will be zero in Equation 5.2, thus, the influence from neighbor j will not be used for updating state of target i.

In either case, $\exists k' \mid \forall l > k'$,

$$p(x_i^{t,l} \mid x_{j \in NN_i}^t, x_i^{t-1}) = p(x_i^{t,k'} \mid x_{j \in NN_i}^t, x_i^{t-1})$$

It follows from Algorithm 2, Line 12 that for

$$0 < \eta_i \leq \left\lceil \frac{\log\left(p(z_i^{t,k'} \mid \hat{x}_i^{t,k'})\right)}{\log \tau} \right\rceil - 1 < \infty$$

state of target i will be updated by Algorithm 2.

□

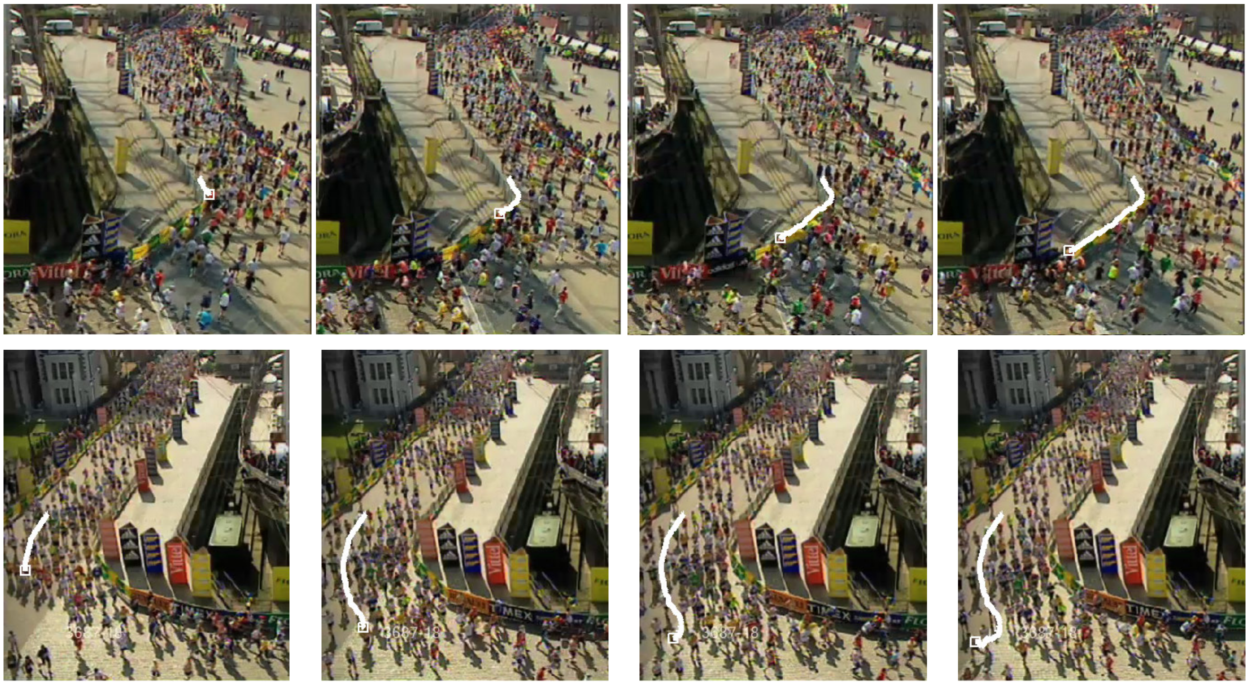


Figure 5.8: Results of Delayed Update: (a) An anomaly where a person is moving against the crowd flow. (b) Result of tracking a particular individual who initially moved with the crowd but later decided to leave the marathon. This shows that instantaneous flow provides reasonable predictions when tracking anomalies. Note that, we do not detect anomalies per se, but whenever the appearance-based confidence from the underlying tracker is below τ, which may happen in the case of anomalies, we rely on instantaneous flow to provide predictions.

Figure 5.7(a) shows the results of hierarchical update for one of the frames in Sequence 2 containing 220 individuals. The order of update is color-coded with bar shown on the right, where red signifies individuals whose state was updated before the ones shown in yellow. The queens are marked with an black square inscribed in a red square. In some instances, yellow squares occur in close proximity highlighting the delayed update where we wait till more neighbors get updated, or update such individuals based on instantaneous flow (Algorithm 2, Line 15). In Figure 5.7(b), we show the final graph produced by the update scheme. The arrows indicate the direction in which the influenced was transmitted. An interesting observation regarding Algorithm 2 is that initially, when updating queens, we do not use any information from neighbors. However, as we move down the hierarchy and away from the queens, we begin to employ more information from neighbors. In this figure, state prediction for several non-queen targets in orange to yellow is influenced by neighbors, which were adjacent to different queens. Therefore, as we move away from queens, the confidence due to prominence subsides, however, it is somewhat compensated by information from an increased number of updated neighbors down the hierarchy.

Relationship to Bayesian Networks The hierarchical order for tracking we propose in this work is similar to belief propagation on a graph with directed edges but no cycles, which is equivalent to directed acyclic graph (DAG), or a Bayesian Network (Figure 5.7(b)). The evidence is provided by the underlying tracker, and our goal is to find an estimate for the positions of all individuals in the crowd given their respective evidence. The conditional probabilities are mixture-of-Gaussians distributions and are provided by NMC. The update scheme starts with the prominent individuals, followed by their neighbors in a one-by-one fashion. Since edges only emanate from nodes (individuals) whose states (positions) have already been updated, the topology of the network evolves and changes till states of all individuals in the scene have been updated. Hierarchical Update can, thus, be seen as a single pass of messages over this time-varying Bayesian Network. An additional advantage of this updating scheme is that it allows handling of anomalous motions, e.g., individuals whose motion does not conform with their neighbors. Since, if the probability of state given local evidence is low for an individual, we ignore the messages received from other individuals and resort to instantaneous flow, which would not have been possible if we used a simultaneous solution. Figure 5.8 provides two instances from Sequence 5 where the proposed method was successfully able to track anomalous individuals whose movement significantly deviated from rest of the crowd.

5.5 Experiments

We tested the proposed method on a variety of sequences which differed in terms of crowd density and tracking difficulty. There are a total of 8 sequences depicting commuters walking outdoors (Sequence $1-2$: high density), marathons with people running at various speeds (Sequence $3-6$: high density), and railway stations (sequence 7: medium density and 8: low density). The first frame of each sequence is shown in Figure 5.9 in the first and third columns, with the corresponding sequence number at bottom-right of the frame. We manually annotated the eight sequences with the total number of individuals annotated in each sequence ranging from $58-747$. Some statistics on the these sequence are shown in the first three rows of Table 5.1.

Although the proposed approach is complementary and an alternative to methods that track dense crowds after modeling crowd flow, we still compare against methods by Ali and Shah [11], who model crowd flow using various floor fields (FF), as well as Rodriguez et al. [30] who use Correlated Topic Model (CTM) to capture crowd behavior. The idea is to ensure that the performance without learning crowd flow remains comparable to the alternative approaches where crowd flow is modeled in advance, i.e., where data from the future is used to learn the behavior of the crowd. In addition, we compared against Park et al. [82] who use contextual information for tracking by solving a MRF framework using mean-shift belief propagation (MSBP). We also generated results from the template-based trackers such as Meanshift (MS) and Normalized Cross Correlation (NCC). NCC was used as the underlying tracker for the proposed method, as given in Equation 5.6.

Both the previous methods [11, 30] track individuals after manually initialization. The primary reason is to discard the effect of human detection which is extremely difficult for these sequences. We also manually initialized individuals by placing a fixed-sized square template around the initialization location. Template size for each sequence is given in the third row of Table 5.1. In addition, new individuals were initialized as they entered the scene. The queens were selected only when new initializations took place, which typically occurred after every fifty frames. The number of clusters for queen selection was set at $k = 100$ for all sequences. The templates were updated after every 10 frames and the value of $\tau = 0.90$ was selected. Therefore, if the value from the underlying tracker at peak location was greater than 0.90, the position of individual was updated. A higher value for this threshold reduced the performance due to increased dependence on instantaneous flow, which is sometimes very noisy.

Figure 5.9 shows the results obtained for the eight sequences. The first and third column show the initial frame of each sequence with all the tracks output by proposed method. In the second and fourth columns, we show graphs that reflect tracking accuracies of various methods. In these graphs, the x-axis shows the distance in pixels ranging from 0 to 25 and the y-axis is the percentage of tracked point from all trajectories that lie within that distance from the corresponding ground truth points. The curve from the proposed method is shown in red. The other methods are MSBP [82] shown in green, FF [11] in yellow, CTM [30] in orange, as well as baseline MS (mean-shift) in cyan and NCC (normalized cross-correlation) in blue. The values of these curves at 15 pixel threshold are given in Table 5.1. The proposed method performs equal or better than the comparison methods for Sequences $1 - 6$. This illustrates that even without learning crowd flow, the prominence and spatial context are helpful enough to give decent tracking results. However, for Sequences 7 and 8, the results are lower by 1 and 2 percents, respectively. For Sequence 7, the reason is primarily the camera angle and large perspective distortion compared to other sequences. For Sequence 8, the reason lies in the density of the crowd. At lower densities, the individuals have more freedom to move, and thus, the motion of neighboring and prominent individuals is not a reliable estimate of the motion of a particular individual under consideration. Furthermore, the evaluation on these two sequences is also done on fewer people than the rest of the sequences.

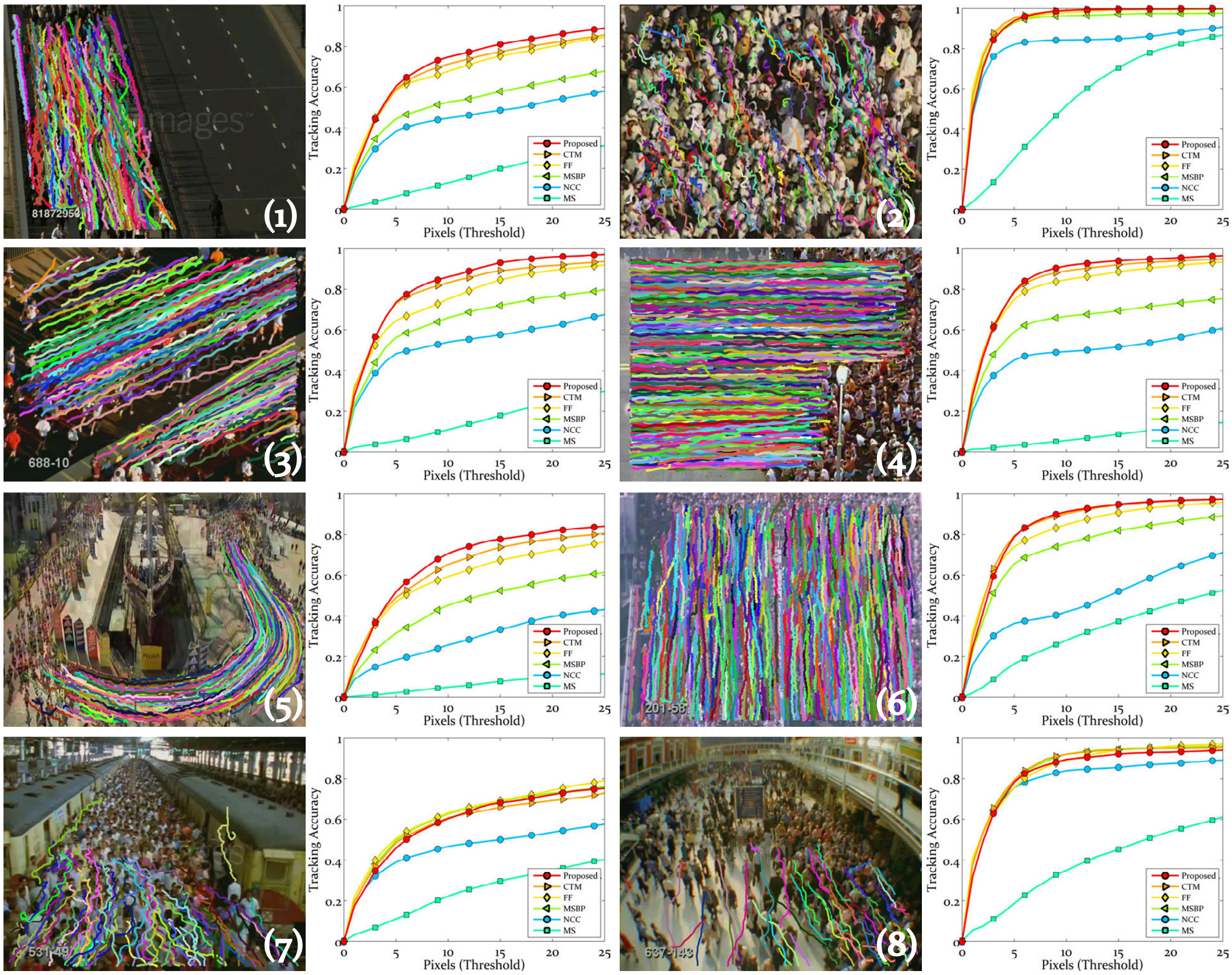


Figure 5.9: Results on eight sequences used in our experiments: Tracks obtained from the proposed method are shown on the first frame of each sequence, shown in first and third columns. Graphs in the second and fourth columns show the tracking accuracies of baseline NCC (blue), MS (cyan), MSBP [82] (green), FF [11] (yellow), CTM [30] (orange), as well as the proposed method (red).

Table 5.1: Quantitative Comparison: Some statistics for the eight sequences are given in first three rows, while the last six rows are the results for the six methods. These are the values of curves in Figure 5.9 at $T = 15$ pixels, which signifies the percentage of points in all tracks that lie within 15 pixels of ground truth. These results show that the proposed method outperforms the comparison methods in most of the sequences.

	Seq 1	**Seq 2**	**Seq 3**	**Seq 4**	**Seq 5**	**Seq 6**	**Seq 7**	**Seq 8**
# Frames	840	134	144	492	464	333	494	126
# People	152	235	175	747	171	600	73	58
Template Size	14	16	14	16	8	10	10	10
NCC	49%	85%	58%	52%	33%	52%	50%	86%
MS	19%	67%	16%	8%	7%	36%	28%	43%
MSBP	57%	97%	71%	69%	51%	81%	**68%**	**94%**
FF	74%	99%	83%	88%	66%	90%	**68%**	93%
CTM	76%	**100%**	88%	92%	72%	**94%**	65%	**94%**
Proposed	**80%**	**100%**	**92%**	**94%**	**77%**	**94%**	67%	92%

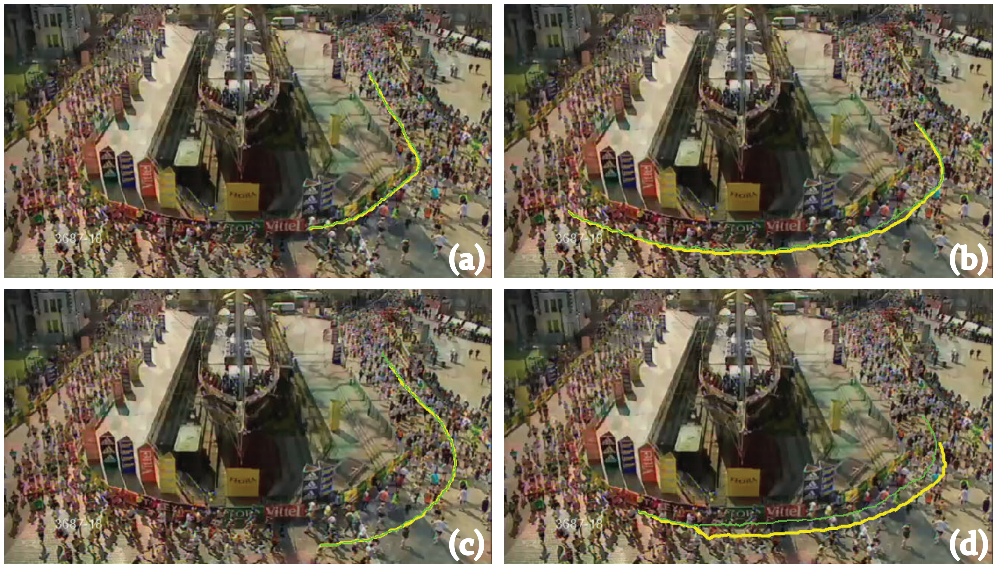


Figure 5.10: Qualitative results on Sequence 5: This figure shows tracks of four different individuals from Sequence 5. The ground truth is shown in green, while track from proposed method is in yellow. In (a-c), the track from proposed method perfectly aligns with the ground truth, while (d) shows a failure case, where the track was lost soon after the initialization.

Next, we present some qualitative results on Sequences 4 and 5. In Figure 5.10, we show tracks of four different individuals from Sequence 5. The ground truth is shown in green, while track from proposed method is in yellow. In Figure 5.10(a-c), the track from proposed method perfectly aligns with the ground truth, while Figure 5.10(d) shows a failure case, where the track was lost soon after the initialization. The reason for this failure was that the person under consideration was wearing a dark-colored shirt. After a few frames of successful tracking, the person came into a position where he or she was surrounded by shadows of several other individuals. The underlying tracker confused the shadow with the person and started chasing the shadow. This highlights the importance of appearance aspect of tracking dense crowds, since it can sometimes dominate auxiliary information provided in the form of better predictions.

Figure 5.11 shows eight examples of tracks obtained on Sequence 4 from the proposed method, FF [11] and CTM [30]. In this figure, the green track is manually-labeled ground truth, while yellow, orange and red tracks correspond to [11], [30] and proposed method, respectively. An analysis of the erroneous tracks reveals that most of the id-switches were between people wearing the same color. The proposed method captures the constraints from neighbors which prohibit the jumping of the tracker across different people. The first row (Figure 5.11(a,b)) shows instances where FF failed to track the individuals, whereas both CTM and proposed method successfully tracked the individuals. The second row (Figure 5.11(c,d)) shows instances where CTM failed, but FF and proposed method were successful. The third row (Figure 5.11(e,f)) shows instances where only the proposed method was successfully able to track the individuals. The last row shows an instance where all methods succeeded (Figure 5.11(g)), and where all failed (Figure 5.11(h)).

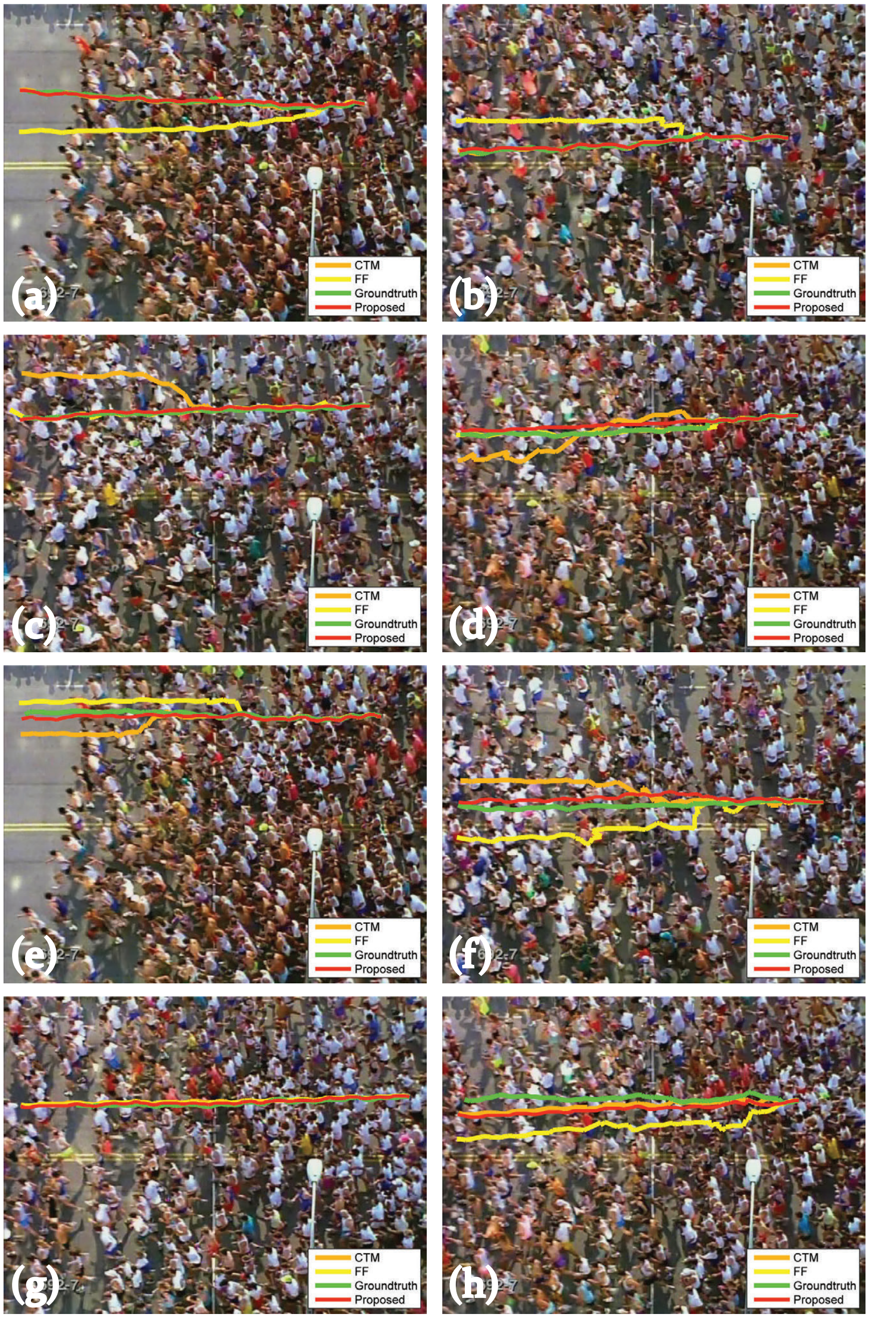


Figure 5.11: Eight examples from Sequence 4 that show the comparison of the proposed method (red) with FF [11] (yellow) and CTM [30] (orange). The ground truth track is depicted in green. (a,b) show instances where FF failed but CTM and proposed method succeeded. (c,d) show instances where CTM failed but the other two succeeded. (e,f) show instances where both FF and CTM failed but the proposed method succeeded. Finally, (g) shows a common instance where all trackers successfully tracked the individual, while (h) shows a rare case where all three failed.

In order to test the contributions of various aspects of the proposed method, we ran a small experiment whose results are presented in Figure 5.12. This plot shows that without the guidance of the queens and neighbors, i.e. using only self-component p_S in Equation 5.1, the results are close to 70%; influence from neighbors, in the form of NMC with randomly initialized queens, adds 20% to tracking accuracy; while salient queens identified using Algorithm 1 add another 6%, giving 96% tracking accuracy of the proposed algorithm at the 10 pixel threshold. For this particular sequence, both prominence and NMC contribute to increase in tracking accuracy, however, this may not always be the case. For instance, prominence is of little value when all people have the same appearance or when everybody in the scene looks different and distinguishable. Similarly, the assumption of motion concurrence breaks at low densities when people have more freedom to move. However, it can be concluded from Figures 5.9 and 5.12 that, while the contributions will vary for different scenes, in general, all components are necessary for an increase in tracking accuracy in structured dense crowds.

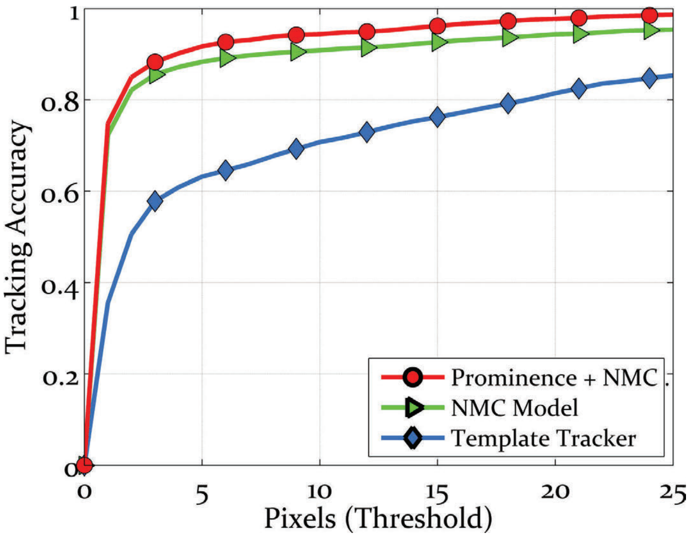


Figure 5.12: Contribution towards tracking accuracy by major components of the algorithm: The experiment was done on sequence 2 at 2.5 fps. The x-axis is the distance threshold in pixels, while the y-axis is the percentage of tracked points that lie within that distance from the ground truth. This shows that all aspects are important for improvement in tracking results.

5.6 Chapter Summary

We introduced a novel method for tracking in dense crowds without using any prior knowledge about the scene, in contrast to previous works which always use some training and modeling of crowd flow using data from the past as well as the future. Beginning with prominent individuals, we track all individuals in the crowd in a ordered fashion employing influence from the neighbors and confidence from template-based tracker. We delay the update of individuals if the confidence from underlying tracker is low. We showed the performance of added functionality via scene-derived visual and contextual information, which significantly improved upon the underlying template-based tracker.

CHAPTER 6: CONCLUSION AND FUTURE WORK

This dissertation addresses several important problems related to dense crowds on a scale not tackled before in literature. Images of extremely dense crowds containing hundreds to thousands of people allow counting and rough approximation of positions of individuals as was addressed Chapter 3. In medium to high density crowds, it is feasible to put exact bounding boxes as was done in our approach for human detection in Chapter 4. And in videos of crowded scenes, tracking of individuals across frames of the video is possible which was addressed in Chapter 5.

Automated analysis of crowds is challenging due to large number of individuals, occlusions, clutter, and fewer pixels per person. To overcome these challenges, we introduced and exploited scene-derived context that is automatically discovered from the crowded scene. For counting, we used multiple sources of information to estimate counts in small patches. These included head detections, Fourier and interest-point based analysis. We captured the contributions of these sources in terms of counts, confidences, and various statistical features which were then used to estimate counts using Support Vector Regression. And since the correct scale while regression cannot be determined in advance, we performed the count estimation at multiple scales and fused the results using a multi-scale Markov Random Field framework. Similarly, we employed repetition of heads for the task of localization where the goal was to approximate the head location of individuals in the crowd image. Initial hypotheses were expanded through multiple criteria and final selection of correct hypotheses was performed using binary integer quadratic programming.

For human detection, we proposed to discover scale and confidence priors of an image using the proposed influence function induced by each detection in its neighborhood. The scores for millions of detection hypotheses were then re-evaluated using the discovered priors. To handle the issue of partial visibility, we proposed to divided the bias term of Latent SVM and used that to construct multiple detectors, each constituting a different combination of parts. The main advantage of this approach was the lack of requirement of annotated ground truth data for training the different detectors. Finally, occlusion reasoning was formulated as a minimization problem for the entire image, and parts of detections were switched on or off based on their scores subjected to overlap and chain constraints which captured occlusion and contiguity, respectively.

The last problem addressed was tracking where context was used in the form of saliency and motion similarity. Prominent or salient individuals were automatically detected from the scene who served as guides for their neighbors. A new motion model - Neighborhood Motion Concurrence - specifically tailored to dense crowds was introduced which captured the similarity of motion exhibited by individuals in dense crowds. A hierarchical tracking framework combined prominence, the motion model as well as confidence from underlying tracker to update positions of individuals at every frame.

In summary, the primary notion explored in this dissertation is that of consistency or smoothness. While counting, we ensured that counts in neighboring patches are similar. This is a valid assumption as the perspective effect introduced by camera is smooth function of distance from the camera. For localization, we employed the idea that a crowded scene is repetitive and appearance of neighboring humans depicted in such an image is consistent and similar. In our approach for human detection, we enforced the scale of humans to be consistent in local neighborhoods. Similarly, we exploited the similarity of motion of neighboring individuals in dense crowds to tackle the problem of tracking. Thus, this dissertation shows that context utilized in the form of local consistencies and similarities is an important cue that can allow us to tackle the challenging problems associated with extremely dense crowds. Furthermore, the two new datasets used to evaluate the performance of our approaches in this dissertation will be useful for other researchers who are working in the area of crowd analysis.

Next, we highlight some extensions and areas of future research that are likely to improve performance of proposed approaches:

Counting. The results are likely to improve through the estimation of ground plane orientation as well as estimation of perspective distortion. The former is applicable to scenarios where perspective distortion is present but estimation may be incorrect. This may happen when orientation of the plane containing the crowd is different from the ground plane, e.g., seating arrangement in a stadium. Furthermore, computation of perspective distortion will allow us to incorporate more robust smoothing prior cues over small spatial regions, and will also allow independent computation of counts for different region sizes. Second, as part of estimation of perspective distortion, it is also important to estimate the line at infinity and the horizon line. This is consequential for counting and density estimation because the non-ground plane regions, such as the sky may also exhibit textures that can be confused with crowd textures or appearance. The estimation of horizon line is a direct way to bypass problems caused by such potential similarity in textures. Third, exploring different feature spaces and descriptors that are likely to be more discriminative in terms of the relative crowd densities they represent is also important. Examples of the disparities between features in representing crowd counts or crowd density include intensity versus first or second derivative of intensity. While appearance, intensity, and color are not a good measure of relative start and end of landmarks representing a unit crowd, the derivatives are likely to be much more useful.

Another idea is to ensure that the regressor or learning algorithms adapts to different regions of input space, instead of learning one hyperplane for the entire input data. This can be implicitly done through Multiple Kernel Learning especially which gives different weights to different kernels (coming from different features). The different regions in the input space can have separate kernels gated using a gating function. In this direction, we propose to explore the effect of different features in different densities of the crowd and how those features can be combined through Multiple Kernel Learning so that counting results improve across all densities.

Human Detection. The proposed approach for human detection consisted of three modules where discovery of scale and confidence priors was coupled with combinations-of-parts detection but global occlusion reasoning was executed independently. Although modularity has its own advantages, it would be an interesting direction to combine all three ideas into one simultaneous solution that also bypasses any post-processing. But, perhaps the most important area of improvement is in human detection itself, where any improvement in the underlying human detector will translate to better performance in dense crowds using the proposed approach.

Tracking. An important constraint for tracking individuals in overhead cameras is the absence of overlap. This can be enforced in the current formulation by ensuring that the bounding boxes of two individuals do not overlap, which is only possible if all individuals in a dense crowd are updated simultaneously instead of the proposed hierarchical tracking approach. Another direction includes making the proposed method more robust by using information from multiple frames at the same time.

LIST OF REFERENCES

[1] “A history of hajj tragedies,” *The Guardian*, 2006. http://www.guardian.co.uk/world/2006/jan/13/saudiarabia. [Accessed: July 1, 2013].

[2] D. Helbing and P. Mukerji, “Crowd disasters as systemic failures: analysis of the love parade disaster,” *EPJ Data Science*, vol. 1, no. 1, pp. 1–40, 2012.

[3] T. E. Board, “Bombs at the marathon,” *The New York Times*, April 15, 2013. http://www.nytimes.com/2013/04/16/opinion/bombs-at-the-boston-marathon.html?ref=bostonmarathon, [Accessed: July 1, 2013].

[4] J. Silveira, J. Junior, S. Musse, and C. Jung, “Crowd analysis using computer vision techniques,” *Signal Processing Magazine, IEEE*, vol. 27, no. 5, pp. 66–77, 2010.

[5] B. Zhan, D. Monekosso, P. Remagnino, S. Velastin, and L. Xu, “Crowd analysis: a survey,” *Machine Vision Applications Journal*, vol. 19, no. 5-6, pp. 345–357, 2008.

[6] B. Zhan, D. Monekosso, P. Remagnino, S. Velastin, and L.-Q. Xu, “Crowd analysis: a survey,” *Machine Vision Applications Journal*, vol. 19, pp. 345–357, 2008.

[7] M. Hu, S. Ali, and M. Shah, “Learning motion patterns in crowded scenes using motion flow field,” in *International Conference on Image Processing*, 2006.

[8] A. Chan and N. Vasconcelos, “Counting people with low-level features and bayesian regression,” *IEEE Transactions on Image Processing*, vol. 21, pp. 2160–77, 2012.

[9] S. Ali and M. Shah, “A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[10] D. Lin, E. Grimson, and J. Fisher, “Modeling and estimating persistent motion with geometric flows,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[11] S. Ali and M. Shah, “Floor fields for tracking in high density crowd scenes,” in *European Conference on Computer Vision*, 2008.

[12] M. Rodriguez, S. Ali, and T. Kanade, “Tracking in unstructured crowded scenes,” in *IEEE International Conference on Computer Vision*, 2009.

[13] L. Kratz and K. Nishino, “Tracking pedestrians using local spatio-temporal motion patterns in extremely crowded scenes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 5, pp. 987–1002, 2012.

[14] R. Melina, “How is crowd size estimated?,” in *Life’sLittleMysteries.com*, 2010.

[15] A. Chan, Z. Liang, and N. Vasconcelos, “Privacy preserving crowd monitoring: Counting people without people models or tracking,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[16] K. Chen, C. Loy, S. Gong, and T. Xiang, “Feature mining for localised crowd counting,” in *British Machine Vision Conference*, 2012.

[17] M. Rodriguez, J. Sivic, I. Laptev, and J. Y. Audibert, “Density-aware person detection and tracking in crowds,” in *IEEE International Conference on Computer Vision*, 2011.

[18] S. Zhu, C. Guo, Y. Wu, and Y. Wang, “What are textons?,” *International Journal of Computer Vision*, pp. 121–143, 2002.

[19] O. Arandjelovic, “Crowd detection from still images,” in *British Machine Vision Conference*, 2008.

[20] R. Azencott, J.-P. Wang, and L. Younes, “Texture classification using windowed fourier filters,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 2, pp. 148–153, 1997.

[21] A. Marana, S. Velastin, L. Costa, and R. Lotufo, “Automatic estimation of crowd density using texture,” in *International Workshop on Signal and Image Processing*, 1997.

[22] T. Leung and J. Malik, “Recognizing surface using three-dimensional textons,” in *IEEE International Conference on Computer Vision*, 1999.

[23] P. Felzenszwalb, D. McAllester, and D. Ramaman, “A discriminatively trained, multiscale, deformable part model,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[24] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part based models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[25] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[26] X. Wang, T. Han, and S. Yan, “An hog-lbp human detector with partial occlusion handling,” in *IEEE International Conference on Computer Vision*, 2009.

[27] P. Dollar, S. Belongie, and P. Perona, “The fastest pedestrian detector in the west,” in *British Machine Vision Conference*, 2010.

[28] Z. Wu, N. Hristov, T. Hedrick, T. Kunz, and M. Betke, “Tracking a large number of objects from multiple views,” in *IEEE International Conference on Computer Vision*, 2009.

[29] X. Song, X. Shao, H. Zhao, J. Cui, R. Shibasaki, and H. Zha, “An online approach: Learning-semantic-scene-by-tracking and tracking-by-learning-semantic-scene,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[30] M. Rodriguez, S. Ali, and T. Kanade, “Tracking in unstructured crowded scenes,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[31] D. Helbing and P. Molnár, “Social force model for pedestrian dynamics,” *Physics Review E*, vol. 51, no. 5, pp. 4282–4286, 1995.

[32] V. Rabaud and S. Belongie, “Counting crowded moving objects,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[33] X. Wu, G. Liang, K. K. Lee, and Y. Xu, “Crowd density estimation using texture analysis and learning,” in *IEEE International Conference on Robotics and Biomimetics*, 2006.

[34] W. Ge, R. Collins, and B. Ruback, “Automatically detecting the small group structure of a crowd,” in *IEEE Workshop on the Applications of Computer Vision*, 2009.

[35] R. Mehran, A. Oyama, and M. Shah, “Abnormal crowd behavior detection using social force model,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[36] L. Kratz and K. Nishino., “Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[37] T. Cao, X. Wu, J. Guo, S. Yu, and Y. Xu, “Abnormal crowd motion analysis,” in *IEEE International Conference on Robotics and Biomimetics*, 2009.

[38] V. B. V. Mahadevan, W. Li and N. Vasconcelos, “Anomaly detection in crowded scenes,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[39] D. Sugimura, K. Kitani, T. Okabe, Y. Sato, and A. Sugimoto, “Using individuality to track individuals: Clustering individual trajectories in crowds using local appearance and frequency trait,” in *IEEE International Conference on Computer Vision*, 2009.

[40] B. Zhou, F. Zhang, and L. Peng, “Higher-order svd analysis for crowd density estimation,” *Computer Vision and Image Understanding*, vol. 116, no. 9, 2012.

[41] J. Ferryman and A. Ellis, “Pets2010: Dataset and challenge,” in *IEEE International Conference on Advanced Video and Signal-Based Surveillance*, 2010.

[42] W. Ge and R. Collins, “Marked point processes for crowd counting,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[43] Z. Zhang, K. Huang, and T. Tan, “Comparison of similarity measures for trajectory clustering in outdoor surveillance scenes,” in *International Conference on Image Processing*, 2006.

[44] G. Brostow and R. Cipolla, “Unsupervised bayesian detection of independent motion in crowds,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[45] X. Wang, X. Ma, and E. Grimson, “Unsupervised activity perception by hierarchical bayesian models,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[46] T. Xiang and S. Gong, “Video behaviour profiling and abnormality detection without manual labelling,” in *IEEE International Conference on Computer Vision*, 2005.

[47] L. Kratz and K. Nishino, “Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[48] S. Cho, T. Chow, and C. Leung, “A neural-based crowd estimation by hybrid global learning algorithm,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 29, no. 4, pp. 535–541, 1999.

[49] D. Kong, D. Gray, and H. Tao, “Counting pedestrians in crowds using viewpoint invariant training,” in *British Machine Vision Conference*, 2005.

[50] D. Ryan, S. Denman, C. Fookes, and S. Sridharan, “Crowd counting using multiple local features,” in *Digital Image Computing: Techniques and Applications*, 2009.

[51] W. Ma, L. Huang, and C. Liu, “Crowd density analysis using co-occurrence texture features,” in *International Conference on Communications and Information Technology*, 2010.

[52] V. Lempitsky and A. Zisserman, “Learning to count objects in images,” in *The Neural Information Processing Systems*, 2010.

[53] P. Dollar, Z. Tu, P. Perona, and S. Belongie, “Integral channel features,” in *British Machine Vision Conference*, 2009.

[54] F. Khan, R. Anwer, J. Weijer, A. Bagdanov, M. Vanrell, and A. Lopez, “Color attributes for object detection,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[55] S. Maji, A. Berg, and J. Malik, “Classification using intersection kernel support vector machines is efficient,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[56] M., Pedersoli, A. Vedaldi, and J. Gonzalez, “A coarse-to-fine approach for fast deformable object detection,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[57] C. Dubout and F. Fleuret, “Exact acceleration of linear object detectors,” in *European Conference on Computer Vision*, 2012.

[58] J. Wu, C. Geyer, and J. Rehg, “Real-time human detection using contour cues,” in *IEEE International Conference on Robotics and Automation*, 2011.

[59] P. Dollar, C. Wojek, B. Schiele, and P. Perona, “Pedestrian detection: An evaluation of the state of the art,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743–761, 2012.

[60] D. Geronimo, A. Lopez, A. Sappa, and T. Graf, “Survey of pedestrian detection for advanced driver assistance systems,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1239–1258, 2010.

[61] B. Wu and R. Nevatia, “Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors,” in *IEEE International Conference on Computer Vision*, 2005.

[62] M. Fink and P. Perona, “Mutual boosting for contextual inference,” in *The Neural Information Processing Systems*, 2003.

[63] K. Mikolajczyk, C. Schmid, and A. Zisserman, “Human detection based on a probabilistic assembly of robust part detectors,” in *European Conference on Computer Vision*, 2004.

[64] Y. Wang, D. Tran, and Z. Liao, “Learning hierarchical poselets for human parsing,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[65] W. Ouyang and X. Wang, “A discriminative deep model for pedestrian detection with occlusion handling,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[66] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. Gavrila, “Multi-cue pedestrian classification with partial occlusion handling,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[67] G. Duan, H. Ai, and S. Lao, “A structural filter approach to human detection,” in *European Conference on Computer Vision*, 2010.

[68] J. Yan, Z. Lei, D. Yi, and S. Li, "Multi-pedestrian detection in crowded scenes: A global view," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[69] W. Ge and R. Collins, "Crowd detection with a multiview sampler," in *European Conference on Computer Vision*, 2010.

[70] L. Kratz and K. Nishino, "Tracking with local spatio-temporal motion patterns in extremely crowded scenes," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[71] X. Zhao, D. Gong, and G. Medioni, "Tracking using motion patterns for very crowded scenes," in *European Conference on Computer Vision*, 2012.

[72] M. Rodriguez, J. Sivic, I. Laptev, and J. Audibert, "Data-driven crowd analysis in videos," in *IEEE International Conference on Computer Vision*, 2011.

[73] P. Carbonetto, N. Freitas, and K. Barnard, "A statistical model for general contextual object recognition," in *European Conference on Computer Vision*, 2004.

[74] A. Torralba, "Contextual priming for object detection," *International Journal of Computer Vision*, vol. 53, no. 2, pp. 169–191, 2003.

[75] D. Ramanan, "Using segmentation to verify object hypotheses," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[76] L. Wolf and S. Bileschi, "A critical view of context," in *International Journal of Computer Vision*, vol. 69, pp. 251–261, 2006.

[77] W. Schwartz, A. Kembhavi, D. Harwood, and L. Davis, "Human detection using partial least squares analysis," in *IEEE International Conference on Computer Vision*, 2009.

[78] D. Hoiem, A. Efros, and M. Hebert, "Putting objects in perspective," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[79] S. Divvala, D. Hoiem, J. Hays, A. Efros, and M. Hebert, “An empirical study of context in object detection,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[80] C. Desai, D. Ramanan, and C. Fowlkes, “Discriminative models for multi-class object layout,” *International Journal of Computer Vision*, vol. 95, no. 1, pp. 1–12, 2011.

[81] Y. Ding and J. Xiao, “Contextual boost for pedestrian detection,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[82] M. Park, Y. Liu, and R. T. Collins, “Efficient mean shift belief propagation for vision tracking,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[83] M. Yang, Y. Wu, and S. Lao, “Intelligent collaborative tracking by mining auxiliary objects,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[84] M. Yang, Y. Wu, and G. Hua, “Context-aware visual tracking,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.

[85] Z. Khan, T. Balch, and F. Dellaert, “An mcmc-based particle filter for tracking multiple interacting targets,” in *European Conference on Computer Vision*, 2004.

[86] R. T. Collins and Y. Liu, “On-line selection of discriminative tracking features,” in *IEEE International Conference on Computer Vision*, 2003.

[87] V. Mahadevan and N. Vasconcelos, “Saliency-based discriminant tracking,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[88] R. A. Smith, “Density, velocity and flow relationships for closely packed crowds,” *Safety Science*, vol. 18, no. 4, pp. 321 – 327, 1995.

[89] Z. Fang, S. M. Lo, and J. A. Lu, “On the relationship between crowd density and movement velocity,” *Fire Safety Journal*, vol. 38, no. 3, pp. 271 – 283, 2003.

[90] M. Moussad, N. Perozo, S. Garnier, D. Helbing, and G. Theraulaz, “The walking behaviour of pedestrian social groups and its impact on crowd dynamics,” *PLoS ONE*, vol. 5, p. e10047, 04 2010.

[91] N. Pelechano, J. M. Allbeck, and N. I. Badler, “Controlling individual agents in high-density crowd simulation,” in *ACM SIGGRAPH/Eurographics symposium on Computer animation*, 2007.

[92] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool, “You’ll never walk alone: Modeling social behavior for multi-target tracking,” in *IEEE International Conference on Computer Vision*, 2009.

[93] K. Yamaguchi, A. C. Berg, L. Ortiz, and T. L. Berg, “Who are you with and where are you going?,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[94] R. Garg, D. Ramanan, S. M. Seitz, and S. N., “Wheres waldo: Matching people in images of crowds,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[95] L. Kratz and K. Nishino, “Going with the flow: Pedestrian efficiency in crowded scenes,” in *European Conference on Computer Vision*, 2012.

[96] L. Leal-Taix, G. Pons-Moll, and B. Rosenhahn, “Everybody needs somebody: modeling social and grouping behavior on a linear programming multiple people tracker,” in *1st ICCV Workshop on Modeling, Simulation and Visual Analysis of Large Crowds*, 2011.

[97] P. F. Felzenszwalb and D. P. Huttenlocher, “Efficient belief propagation for early vision,” *International Journal of Computer Vision*, vol. 70, no. 1, 2006.

[98] W. Ouyang and X. Wang, “Single-pedestrian detection aided by multi-pedestrian detection,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[99] W. Ouyang and X. Wang, “Joint deep learning for pedestrian detection,” in *IEEE International Conference on Computer Vision*, 2013.