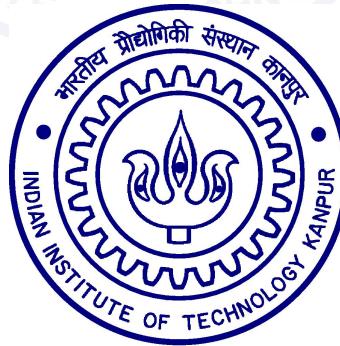
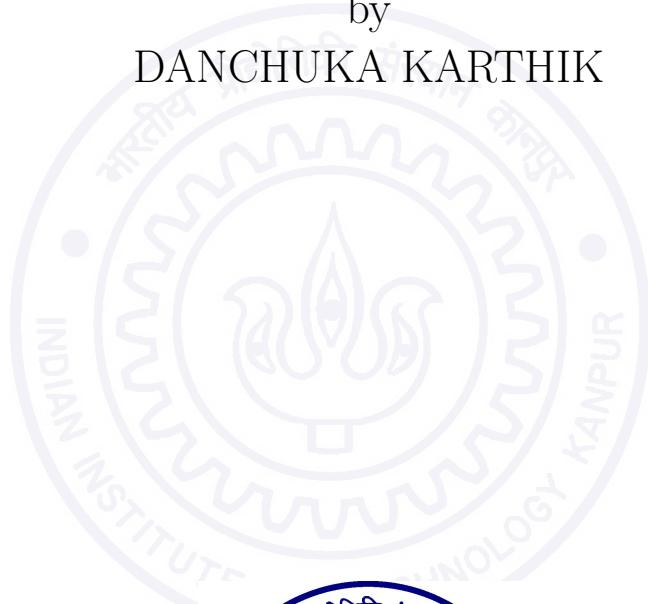


From Crowd Dynamics to Crowd Safety

A Thesis Submitted
in Partial Fulfillment of the Requirements
for the Degree of
Master of Technology

by
DANCHUKA KARTHIK



to the
Department of Electrical Engineering
INDIAN INSTITUTE OF TECHNOLOGY KANPUR
Kanpur, INDIA - 208016
May 2016

to Lord Chaitanya
my eternal Lord,
who propounded divine music, in the form of Sankirtana,

*hare krṣṇa hare krṣṇa, krṣṇa krṣṇa hare hare
hare rāma hare rāma, rāma rāma hare hare*

as the way to obtain Love of God
and revealed to us the beauty of the spiritual world,

kathā gānam nātyam gamanam api vamśī priya-sakhi
– *Sri Brahma-Samhitā 5.56*

where every word is a song, every step is a dance
and flute is the favourite companion.

Certificate

This is to certify that the work contained in the thesis entitled "**From Crowd Dynamics to Crowd Safety**", by DANCHUKA KARTHIK (Roll No.13104036), has been carried out under our supervision for the partial fulfillment of M.Tech degree in the Department of Electrical Engineering, IIT Kanpur and this work has not been submitted elsewhere for any other degree.

K.S.Venkatesh

Professor

Department of Electrical Engineering

Indian Institute of Technology Kanpur

Kanpur, INDIA - 208016

May 2016

Laxmidhar Behera

Professor

Department of Electrical Engineering

Indian Institute of Technology Kanpur

Kanpur, INDIA - 208016

Acknowledgements

Firstly, I would like to express my heartfelt gratitude to my guide, Prof.K.S.Venkatesh and Prof. L. Behera for their valuable guidance and support throughout my research. While he gave me a great degree of freedom in my research, his deep insights into the fundamentals of each problem inspired ideas which ultimately took the shape of this thesis. He has set before me high standards of integrity and mature vision.

I owe my sincere gratitude to my real family – the Bhaktivedanta Club – where I came in contact with the teachings of His Divine Grace A. C. Bhaktivedanta Swami Srila Prabhupada and got the shelter of my spiritual master His Holiness Gopal Krishna Goswami. I am indebted to Sir who taught me that the purpose of all our endeavours should be to purify our experience. I am highly obliged to all the devotees of Bhaktivedanta Club, in particular Mataji and Vipul Pr and Tharun Pr for being so kind to me for all these years. Their jubilant Kirtans and loving association teaches me the true meaning of spiritual life.

I would like to take the opportunity to express my deep gratitude to the esteemed faculty members of IIT Kanpur, who nurtured me. I would like to thank all my lab mates and especially my seniors for guiding me and extending me all help whenever I needed it.

Last but not the least, I would like to thank my parents, and my brother for their encouragement, patience and love.

Abstract

At present there are large number of Crowd gatherings all over the world but also there are numerous crowd disasters occurring every year. Unfortunately, the information about the (spatio-temporal) development of these events tend to be qualitative rather than a systematic quantitative analysis, and there are not many preventive measures to manage the crowd and control the huge crowd during the critical crowd conditions in order to avoid Crowd Disaster. During Crowd gathering we can collect “live video recordings”. With this video material, it is possible to analyse the behaviour of the crowd. Further more, we may use this analysis to manage the crowd movement as well as to avoid the crowd Disaster.

This thesis aims at analyzing the crowd by using live video recording and use the crowd analysis to manage the crowd and prevent the disaster. First, a novel mathematical crowd model is proposed where we have re-defined the crowd parameters such as Local Density, Local Velocity, Variation in Velocity, Pressure which play a crucial role in analyzing the crowd as well as managing the crowd to avoid the crowd disasters. Further, in order to implement the crowd model we proposed a novel method to detect the head in the given location which is an important step in crowd modeling. In this method we have cascaded the HOG with the local minimum and maximum which has increased our speed of detection without reducing the detection rate. Finally, we have implemented the crowd model incorporating head detection information by which we are able to analyze the density, velocity, critical regions all of whom can be used for studying the behavior of crowd, manage the flow of the crowd to avoid the Crowd Disasters.

Contents

Acknowledgements	vii
Abstract	ix
1 Introduction	1
1.1 Background	1
1.2 Project Motivation	2
1.3 State of the Art	3
1.3.1 Pedestrians detection	4
1.3.2 Crowd Behaviour	5
1.4 Application	6
1.5 Thesis Outline	7
2 Head Detection	9
2.1 HOG Features :	9
2.2 Support Vector Machine based Classification	12
2.3 Sliding Window based detection	15
2.4 Estimating head by Local Maxima/Local Minima in a image	16
2.5 HOG Feature with local maxima/minima	17
2.6 Result of Head Detection	18
3 Unsupervised deep learning models for Head detection	25
3.0.1 Two basic kinds of Machine Learning	25
3.0.2 Deep learning	26
3.0.3 Biological Motivation	26
3.0.4 Deep Belief networks	27
3.0.5 Gradient of the Log-likelihood	29
3.0.6 Experiments	31
4 Crowd Modeling	35
4.1 Crowd Behavior	35
4.1.1 Microscopic Approach :	35

4.1.2	Macroscopic Approach :	36
4.1.3	Lane Formation	36
4.1.4	Stripe Formation	37
4.1.5	Pedestrian Trail Formation	37
4.1.6	Bottleneck	38
4.1.7	Directional Bottleneck Flows	39
4.1.8	Stop-and-Go Waves	39
4.1.9	Crowd Turbulence	40
4.2	Crowd Model	41
4.2.1	Parameters of Crowd Model	42
4.2.2	View Point Variation of Video Dataset	42
4.2.3	Local Density	44
4.2.4	Results of Density	49
4.2.5	Local Velocity	50
4.2.6	Results of Velocity by kalman tracking	50
4.2.7	Variation in Velocity	52
4.2.8	Pressure	53
4.2.9	Results of Pressure	54
5	Conclusion and Future Scope	57
5.1	Conclusion	57
5.2	Future Scope	58
5.3	Some Measures to Improve Crowd Safety	59

List of Figures

2.1	Histogram of Orientation (HOG) Feature	11
2.2	training data	13
2.3	Support Vector Machine	13
2.4	Original Image	18
2.5	Local Maxima and Minima of Image	19
2.6	Watershed segmentation of Local Maxima and Minima Image	20
2.7	Finding Center of Segmentation	21
2.8	Finding local max of image	21
2.9	Estimating the head by local maxima	22
2.10	Filtering by Hog Feature	22
2.11	Head Detection by Hog feature	23
2.12	Detection Vs frame	23
3.1	Restricted Boltzmann machine	27
3.2	Deep Belief Network	30
3.3	Deep neural network (DBN+output layer)	31
3.4	A group of people	32
3.5	Reconstruction error	33
3.6	Training error	33
4.1	Crowd Model	42
4.2	Perpendicular View of Camera	43
4.3	Perspective View of Camera	44
4.4	High Altitude View of Camera	45
4.5	Voronoi Distribution Based Local Density	46
4.6	Gaussian Distribution based Local Density	47
4.7	Error in Global Density ϱ_G , Local Density ρ_G Vs σ	49
4.8	Global Density Vs frame	49
4.9	Kalman Tracking	50
4.10	Visual Results of pressure	54
4.11	reference Pressure Vs frame	54
4.12	Pressure vs frame	55

5.1	Large View	59
5.2	Large Scale Estimate	60



Chapter 1

Introduction

1.1 Background

Crowd analysis in general has been studied for more than a century. For example, Le Bon [28] investigated the crowds during the French Revolution in the 1890s. This rather qualitative study of crowd psychology was later followed by Mintz [33] who conducted an experiment on escape panics, which was later put into a game theoretic framework by Brown [9]. Both Le Bon and Brown pointed out that the collective behavior can't be explained by the average of actions that the individuals would take individually. Rather, it is something that emerges. Still today, these lines of thought are relevant and are being investigated with quantitative methods, such as computer simulations and analysis of detailed empirical data from experiments or from real-world observations.

Today, the study of pedestrian crowds is an interdisciplinary field, and is carried out within different communities, with different focuses, e.g.:

- Physicists (or so-called socio-physicists) are focusing on explaining self-organization phenomena in pedestrian crowds.
- Fire engineers are simulating evacuation scenarios, to assess the impact of fires and the building geometries on the evacuation time.
- Biologists and social scientists are investigating how pedestrians interact with each other.
- Computer scientists are setting up large-scale multi-agent simulations (often distributed over many computing nodes), as well as making realistic crowd animations.
- Computer scientists in the field of computer vision use videos of pedestrian movements in order to develop algorithms for identifying and tracking the

pedestrians.

- Psychologists are interested in the decision-making mechanisms pedestrians are facing.
- Transportation engineers focus on the mobility of crowds, particularly on the relation of the flow of pedestrians and the crowd density, which is called the fundamental diagram.

The list above is of course a crude generalization, since there are many overlaps between the fields. What can be observed for example from citation and conferences, is that there are many sub-communities within the wide field of pedestrian research.

Sadly, there has not been enough effort in connecting the results of all these sub-communities into one consistent theory. There have been a few studies in the past which are working across many of these sub-fields [41] [34], and this thesis is aiming in that direction

1.2 Project Motivation

The study of crowd dynamics is interesting because of the various self-organization phenomena resulting from the interactions of many pedestrians, which may improve or obstruct their flow. Besides formation of lanes of uniform walking direction and oscillations at bottlenecks at moderate densities, it was recently discovered that stop-and-go waves Helbing et al [26]., and a phenomenon called crowd turbulence can occur at high pedestrian densities Helbing et al[26]. Although the behavior of pedestrian crowds under extreme conditions is decisive for the safety of crowds during the access to or exit from mass events as well as for situations of emergency evacuation, there is still a lack of empirical studies of extreme crowding. Therefore, this thesis discusses how one may study high-density conditions based on suitable video data.

The part of the work is based on Helbing et al. [26] where pedestrian are entering the Jamarat Bridge in Mina, 5 kilometers from the Holy Mosque in Makkah, Saudi-Arabia. The results reveal unexpected pattern formation phenomena and show that the average individual speed does not go to zero even at local densities of 10 persons per square meter. Since the maximum density and flow are different from measurements in other countries, this has implications for the capacity assessment and dimensioning of facilities for mass events. When conditions become congested, the flow drops significantly, which can cause stop-and-go waves and a further increase of

the density until critical crowd conditions is reached. Then, crowd turbulence sets in, which may trigger crowd disasters. For this reason, our thesis aims to operate pedestrian facilities sufficiently below their maximum capacity and to take measures to improve crowd safety some of which are discussed.

1.3 State of the Art

Crowd Dynamics and Analysis is a new and wide area of interest in the research community which could potentially lend itself to a wide range of domain and different applications in different fields where computer vision and image processing plays a key role.

In regard to Crowd Dynamics and Analysis different wide rage of application such as pedestrians detection, estimating Count of pedestrians, tracking, crowd modeling ,abnormal behavior Detection, Crowd Density Estimation, Behaviors of Crowd Scenes etc. Here Pedestrian Detection and Crowd Behavior plays important role and are discussed below

1.3.1 Pedestrians detection

We review pedestrian detectors with a focus on sliding window approaches. These appear most promising for low to medium resolution settings, under which segmentation [17] or keypoint [29], [39] based methods often fail.

Foreground based methods: in Refs. [46], [18], the foreground is extracted firstly by background removal using a reference image, then crowd density is computed as a function of the number of foreground pixels; the function itself is obtained by curve fitting. However, these methods may fail when the background changes gradually over time.

Papageorgiou et al. [13] proposed one of the first sliding window detectors, applying support vector machines (SVM) to an over-complete dictionary of multiscale Haar wavelets. Viola and Jones [VJ] [43] built upon these ideas, introducing integral images for fast feature computation and a cascade structure for efficient detection, and utilizing AdaBoost for automatic feature selection. These ideas continue to serve as a foundation for modern detectors.

Large gains came with the adoption of gradient-based features. Inspired by SIFT lowe2004distinctive, Dalal and Triggs [HOG] [7] popularized histogram of oriented gradient (HOG) features for detection by showing substantial gains over intensity based features. Zhu et al. [50] sped up HOG features by using integral histograms [35].

In earlier work, Shashua et al. [40] proposed a similar representation for characterizing spatially localized parts for modeling pedestrians. Since their introduction, the number of variants of HOG features has proliferated greatly with nearly all modern detectors utilizing them in some form. Shape features are also a frequent cue for detection. Gavrila and Philomin [15], [14] employed the Hausdorff distance transform and a template hierarchy to rapidly match image edges to a set of shape templates. Wu and Nevatia [44] utilized a large pool of short line and curve segments, called edgelet features to represent shape locally. Boosting was used to learn head, torso, leg and full body detectors; this approach was extended in [45] to handle multiple viewpoints. Similarly, shapelets [38] are shape descriptors discriminatively learned from gradients in local patches; boosting was used to combine multiple shapelets into an overall detector [SHAPELET]. Liu et al. [31] proposed granularity-tunable features that allow for representations with levels of uncertainty ranging from edgelet to HOG type features; an extension to the spatio-temporal domain was developed in [30].

1.3.2 Crowd Behaviour

Crowd Behavior is important factor in monitoring the crowd. In Crowd analysis: survey [48] a review of the latest research trends and approaches from different research communities is provided. There are two main approaches in solving the problem of understanding crowd behaviors. In this object based approach, considers a collection of individuals [42],[32]. Therefore, to understand the crowd behavior it is necessary to perform segmentation or detect objects to analyze group behaviors [5]. This approach faces considerable complexity in detection of objects, tracking trajectories, and recognizing activities in dense crowds where the whole process is affected by occlusions.

Other is holistic approaches [3] [1] which consider the crowd as a global entity in analysis of medium to high density scenes. In related works by Avidan et al. in [37] and Chan and Vasconcelos in [6], instead of tracking individual objects, scene modeling techniques are used to capture features for the crowd behavior and car traffic respectively.

These are (holistic) approaches which directly tackle the problem of dense occluded crowds in contrast to the object based methods. In addition, there are some works that mix the bottom-up view of object-based methods with top-down methods such as Ali and Shahs [2] for tracking humans in very dense crowds. Meanwhile, crowd behavior analysis has been an active research topic in simulation and graphic fields where the main goal is to create realistic crowd motions. The real crowd motion exhibits complex behaviors like line forming [24], laminar and turbulent flow

[20][47], arching and clogging at exits, jams around obstacles [22], and panic [20]. Exact simulation of a crowd using behavior modeling leads to design of proper public environments that minimize the possibility of the hazardous events. Furthermore, in the graphics community, it leads to accurate modeling.

1.4 Application

The steady population growth, along with the worldwide urbanization, has made the crowd phenomenon more frequent. It is not surprising, therefore, that crowd analysis has received attention from technical and social research disciplines. The crowd phenomenon is of great interest in a large number of applications:

- **Crowd management:** Crowd analysis can be used for developing crowd management strategies, especially for increasingly more frequently and popular events such as sport matches, large concerts, public demonstrations and so on, to avoid crowd related disasters and insure public safety.
- **Public space design:** Crowd analysis can provide guidelines for the design of public spaces, e.g. to make the layout of shopping malls more convenient to costumers or to optimize the space usage of an office.
- **Virtual environments:** Mathematical models of crowds can be employed in virtual environments to enhance the simulation of crowd phenomena, to enrich the human life experience.
- **Visual surveillance:** Crowd analysis can be used for automatic detection of anomalies and alarms. Furthermore, the ability to track individuals in a crowd could help the police to catch suspects.
- **Intelligent environments:** In some intelligent environments which involve large groups of people, crowd analysis is a pre-requisite for assisting the crowd or an individual in the crowd. For example, in a museum deciding how to divert the crowd based on the patterns of crowd.

Crowd management and public space design are studied by sociologists, psychologists and civil engineers; virtual environments are studied by computer graphic researchers; visual surveillance and intelligent environments are of interest to computer vision researchers. The approach favored by psychology, sociology, civil engineer and computer graphic research is an approach based on human observation and analysis. Sociologists, for instance, study the characters of a crowd as a social phenomenon, exploring human factors.

1.5 Thesis Outline

The present work is organized into five chapters. In the present chapter, a general Introduction of various concepts used in the present work is given. A brief literature survey is also presented.

2nd chapter deals with Object Detection where head is detected by HOG Features

3rd chapter deals with the Unsupervised deep learning models for Head detection

4th chapter deals with Crowd Modeling where a detailed description of the Crowd Model and different parameters like Local Density, Local Velocity, Pressure are presented.

5th chapter ends with conclusion and future work.



Chapter 2

Head Detection

In order to implement a crowd model 4.1, the most important part is finding location of pedestrian $\vec{X}(t)$ in a given dataset. Since the dataset is collected at a height more than the height of pedestrian we assume that the location of the pedestrian is the location of the Head of the pedestrian. Our assumption is made so that we can use object detection.

2.1 HOG Features :

In object detection there are different features trained by different classifiers. HOG feature [7] is one of the popular features used in the object detection.

As we are implementing the model 4.1 by finding the head location of pedestrians $\vec{X}(t)$ we use find different methods and one of most popular method is Hog [7]

In this method Hog feature is extracted from the image and given to the Classifier. We are using a SVM classifier.

Histogram of Oriented Gradient descriptor or HOG descriptor is a feature based descriptor used in computer vision and image processing for object detection. The technique counts occurrences of gradient orientation in localized portions of an image - detection window, or region of interest (ROI).

HOG features have been introduced by Navneet Dalal and Bill Triggs [7] who have developed and tested several variants of HOG descriptors, with differing spatial organization, gradient computation and normalization methods.

The essential thought behind the Histogram of Oriented Gradient descriptors is that local object appearance and shape within an image can be described by the

distribution of intensity gradients or edge directions.

The implementation of these descriptors can be achieved by 2.1

1. computing gradient
2. Orientation Bins and cells dividing
3. Descriptor Blocks
4. Block Normalization

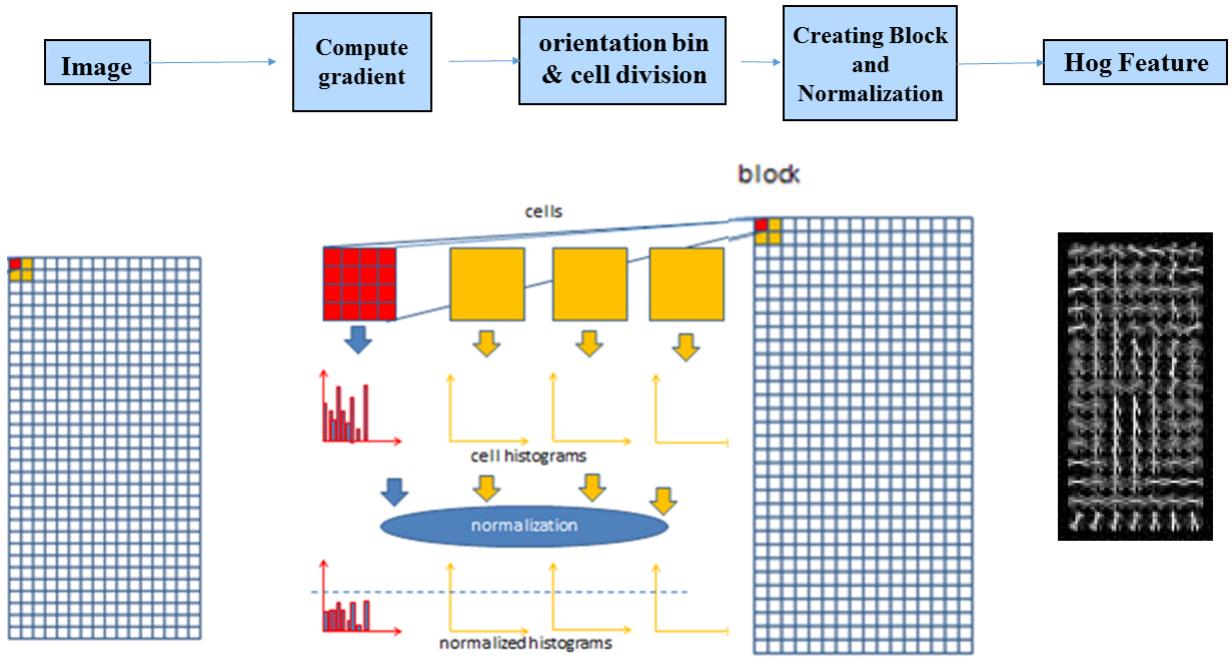


Figure 2.1: Histogram of Orientation (HOG) Feature

Gradient Computation : The first step of calculation is the computation of the gradient values. The most common method is to apply the 1D centered point discrete derivative mask in both the horizontal and vertical directions. Specifically, this method requires filtering the grayscale with the filter kernel .

$$D_x = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \text{ and } D_y = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} D_y = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$$

So, being given an image I , we obtain the x and y derivatives using a convolution operation: $I_x = I * D_x$ and $I_y = I * D_y$

The magnitude of the gradient is $|G| = \sqrt{I_x^2 + I_y^2}$

The orientation of the gradient is given by: $= \arctan \left(\frac{I_y}{I_x} \right)$

2. Orientation Bins and cells dividing In the second step, the image is divided into cells. The size of a cell can be varied and we can calculate the cell histograms. Each pixel within the cell casts a weighted vote for an orientation-based histogram channel based on the values found in the gradient computation. The cells themselves are rectangular and the histogram channels are evenly spread over 0 to 180 degrees or 0 to 360 degrees, depending on whether the gradient is unsigned or signed. Dalal and Triggs [7] found that unsigned gradients used in conjunction with 9 histogram channels performed best in their experiments. As for the vote weight, pixel contribution can be the gradient magnitude itself, or the square root or square of the gradient magnitude.

3. Descriptor Blocks In order to account for changes in illumination and contrast, the gradient strengths must be locally normalized which requires grouping the cells together into larger, spatially-connected blocks. The HOG descriptor is then the vector of the components of the normalized cell histograms from all of the block regions. These blocks typically overlap meaning that each cell contributes more than once to the final descriptor. Two main block geometries exist: rectangular R-HOG blocks and circular C-HOG blocks. R-HOG blocks are generally square grids, represented by three parameters: the number of cells per block, the number of pixels per cell, and the number of channels per cell histogram.

4. Block Normalization There are different methods for block normalization. Let v be the non-normalized vector containing all histograms in a given block, $\|v_k\|$ be its k-norm for $k = 1, 2$ and e be some small constant (whose value will not influence the results). Then the normalization factor can be one of the following:

$$L2 - norm : f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}}$$

$$L1 - norm : f = \frac{v}{\|v\|_1 + e}$$

$$L1 - sqrt : f = \sqrt{\frac{v}{\|v\|_1 + e}}$$

Training Binary classifier by HOG features

We Trained Hog feature containing 3023 training and 1228 test heads of 25 x 25

pixels as shown in figure 2.2. Generally there are wide range of classifiers such as ann, linear classifier etc. We use svm classifier to train HOG feature which is one of the popular binary classifier. The description of the svm is given in [?]

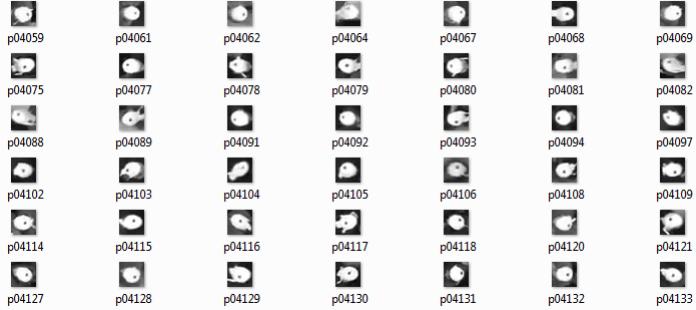


Figure 2.2: training data

2.2 Support Vector Machine based Classification

Support vector machines (SVMs) is a supervised learning method used for classification, regression and outliers detection.

Given a set of training examples consisting of two classes $\{x^+ x^-\}$, then SVM constructs a hyperplane $w^T x + b = 0$ as shown in figure 2.3 which can be used for classification. The hyperplane is constructed in such a way that it has maximum margin (optimal margin) between two classes. In order to construct optimal margin hyperplane we convert hyperplane equation into an optimization problem maximizing the distance between two planes which is referred to as a linear kernel base SVM classifier

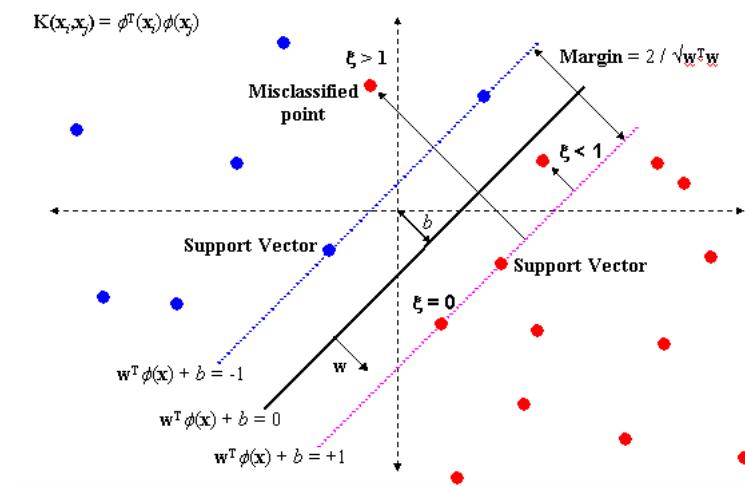


Figure 2.3: Support Vector Machine

Lets introduce the notation used to define formally a hyperplane:

$$f(x) = w^T x + b \quad (2.1)$$

where w is known as the weight vector and b is the bias.

The optimal hyperplane can be represented in an infinite number of different ways by scaling of w and b . As a matter of convention, among all the possible representations of the hyperplane, the one chosen is

$$|w^T x + b| = 1 \quad (2.2)$$

where x symbolizes the training examples closest to the hyperplane.

In general, the training examples that are closest to the hyperplane are called support vectors. This representation is known as the canonical hyperplane. Now, we use the result of geometry that gives the distance between a point x and a hyperplane (w, b) :

$$\text{distance} \quad d = \frac{|w^T x + b|}{\|w\|} \quad (2.3)$$

In particular for the canonical hyperplane, the numerator is equal to one and the distance to the support vectors is

$$d_{\text{support vector}} = \frac{|w^T x + b|}{\|w\|} = \frac{1}{\|w\|} \quad (2.4)$$

Recall that the margin introduced in the previous section, here denoted as d_{margin} , is twice the distance to the closest examples:

$$d_{\text{margin}} = \frac{2}{\|w\|} \quad (2.5)$$

Finally, the problem of maximizing d_{margin} is equivalent to the problem of minimizing a function $L(w)$ subject to constraint for all the training examples x_i given as

$$\min_{w,b} L(w) = \frac{1}{2} \|w\|^2 \quad \text{s.t.} \quad y_i (w^T x_i + b) \geq 1 \quad \forall i, \quad (2.6)$$

where y_i represents each of the labels of the training examples.

This is a problem of Lagrangian optimization that can be solved using Lagrange multipliers to obtain the weight vector w and the bias b of the optimal hyperplane.

The advantages of support vector machines are:

1. Effective in high dimensional spaces.
2. Still effective in cases where number of dimensions is greater than the number of samples.
3. Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.
4. Versatile: Different Kernel functions can be specified for the decision function. Commonly kernels are provided, but it is also possible to specify custom kernels.

The disadvantages of support vector machines include:

1. If the number of features is much greater than the number of samples, the method is likely to give poor performances.
2. SVM's do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation.

2.3 Sliding Window based detection

The above SVM classifies the image of fixed window size(detection window). In order to classify high resolution image containing head at different location we use sliding window where a window is moved by a step size 's' from starting point of image to ending point of the image.

Therefore by using sliding window we detect head in high resolution image. The disadvantage of sliding window is if the window size is large then the search range(no of windows to be classified) is less but for our head detection the head size is small so our search range is very large. Due to this our detection rate for high resolution is very low.

In order to increase the detection rate of the image we reduce the search range by estimating head as keypoints.

2.4 Estimating head by Local Maxima/Local Minima in a image

As discussed above, the computation cost of the HOG based detection using sliding window is very high and in order to reduce the computation cost we are trying

to estimate head location by assuming that each head in a image is either a local maxima or a local minima.

In order to find the local maxima/minima in a image we use difference between intensity of pixel and average of the intensity of neighborhood pixels. The above method is similar to that of keypoint found by Difference of Gaussian(DoG) and Laplacian of Gaussian(LoG).

But by using Gaussian Blur we observe there are disadvantages as described below:

1. We notice that due to gaussian blur in the adjacent heads, some of the heads are mixed in the blur so local maxima/minima is lost and heads are not detected.
2. If salt and pepper noise are present in the given image due to gaussian blurring, the noise is converted into local maxima/minima which gives false head detection.
3. When two heads are overlapping with each other due to gaussian blurring the two local maxima/minima are converged into single local maxima/minima.

Instead of calculating difference between the gaussian filters of an image we use difference between median filters of an image as shown in figure 2.5.

The advantage of difference in median filters is that it preserves the local maxima/minima in the image even when the local maxima/minima are overlapping with each other which is not possible while using gaussian filters. If the intensity of the noisy local maxima/minima is more, then gaussian filter will give noisy local maxima/minima while a difference in median filters removes the noise local maxima/minima whose size is less compared to that of the head size.

Using median filter may increase computational cost by some amount but it is less when compared to that of Hog feature based sliding window detection computation. To locate the local maximum in the given image we use the waterfall based segmentation of difference of median filter as shown in figure 2.6.

The results of head detection are given below:

The advantage of this method is that all heads are detected although there may be some false head detections in our method. The false detection can be reduced by giving the detection results to SVM classifier as discussed above, which will increase our performance of the system.



Figure 2.4: Original Image

2.5 HOG Feature with local maxima/minima

In this section we cascade the local mixima/minima with hog feature based SVM classifier. From the head detected by local maxima/minima the hog features are extracted to classify whether it is the head or non-head. Using this method we are able to reduce the computational cost of HOG based detection. The result are shown in figure 2.9 – 2.10

2.6 Result of Head Detection

As we discussed in HOG [7] the results implemented by using sliding window are given below. Since our input is video we are having ground truth value and detection results for every frame are shown in figure 2.11 and the overall results detection for all the frames is given in figure 2.12. For the graph 2.12 we calculated the average detection detection rate as 98.4 %.

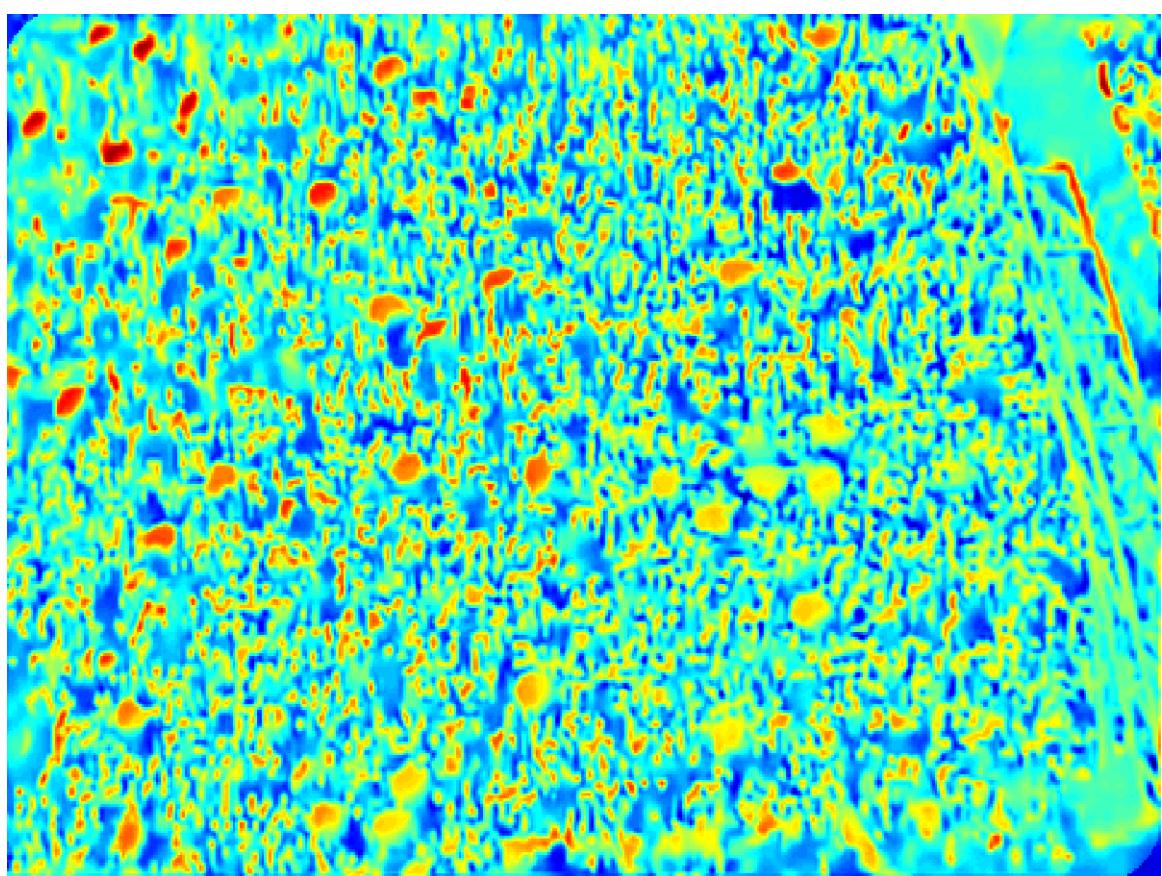


Figure 2.5: Local Maxima and Minima of Image

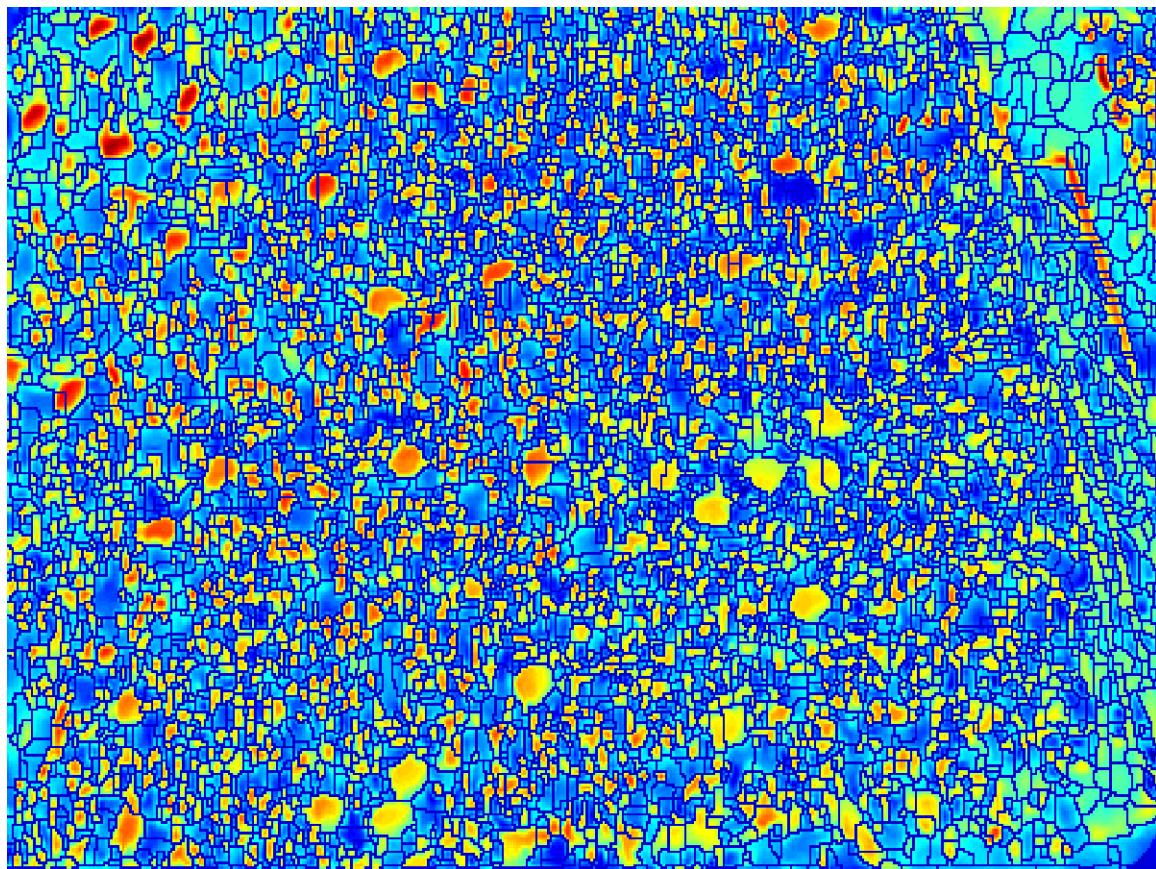


Figure 2.6: Watershed segmentation of Local Maxima and Minima Image

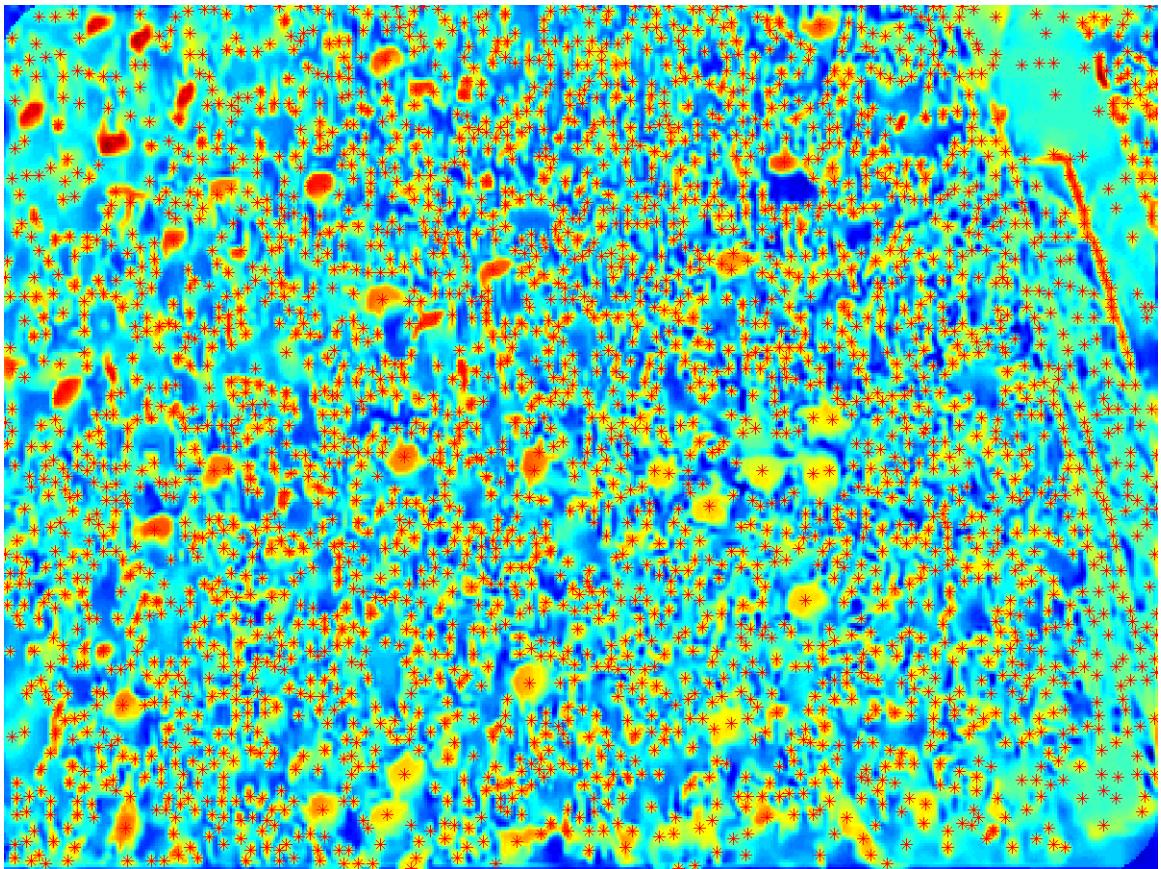


Figure 2.7: Finding Center of Segmentation

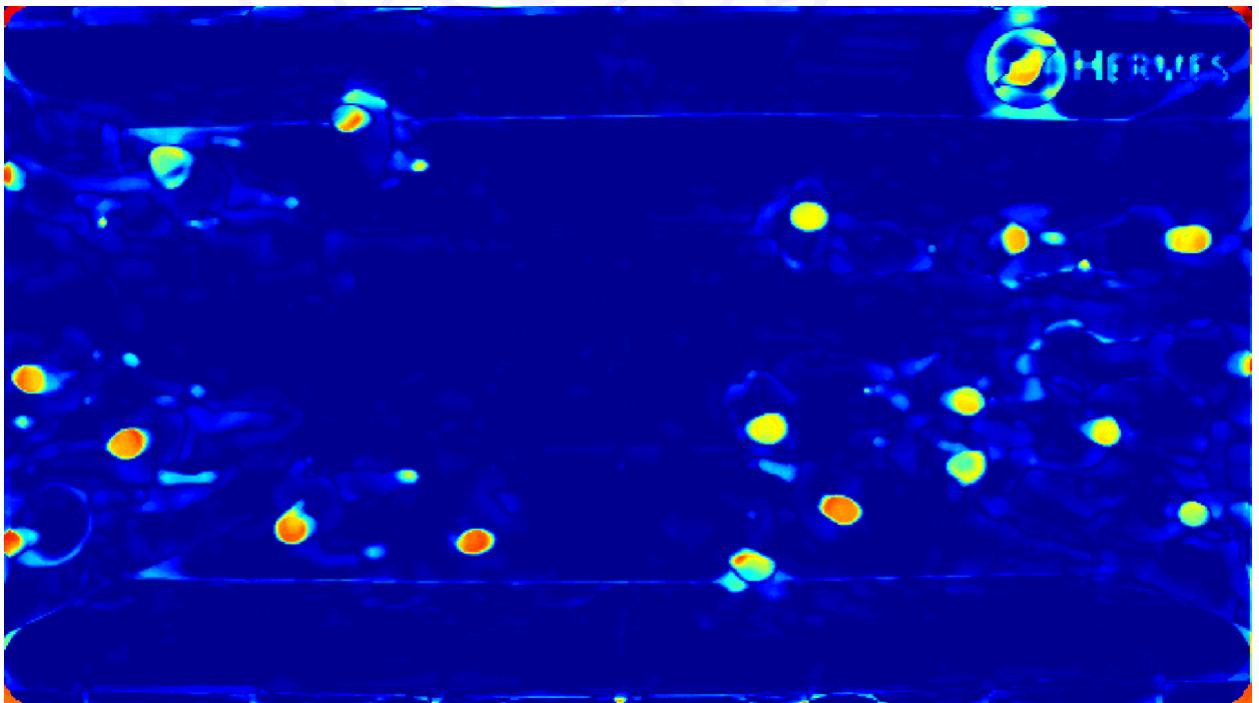


Figure 2.8: Finding local max of image



Figure 2.9: Estimating the head by local maxima

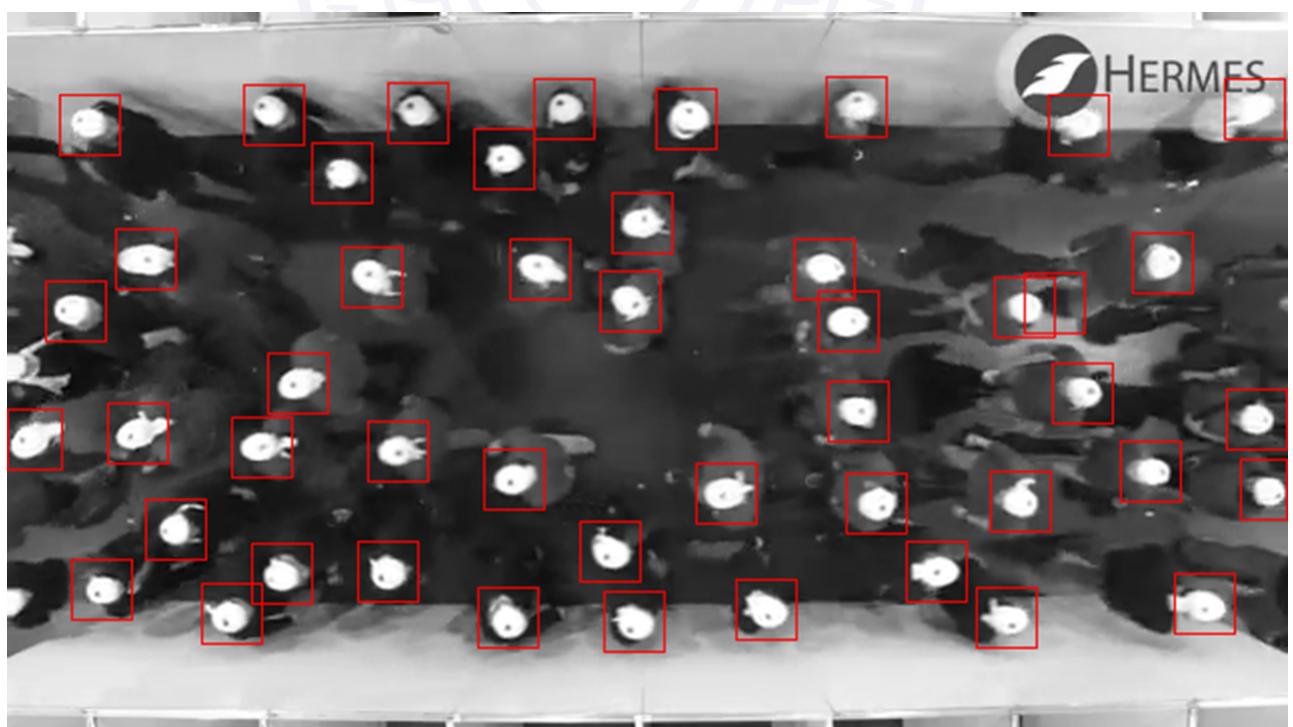


Figure 2.10: Filtering by Hog Feature

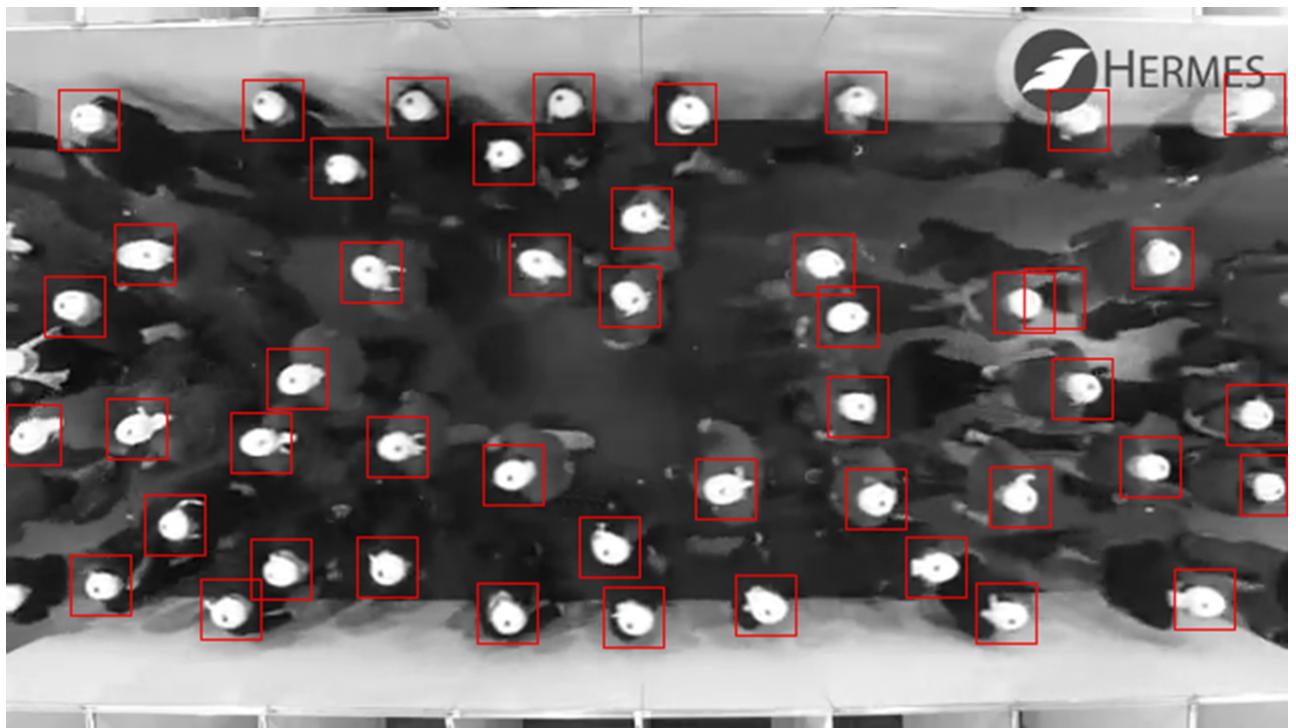


Figure 2.11: Head Detection by Hog feature

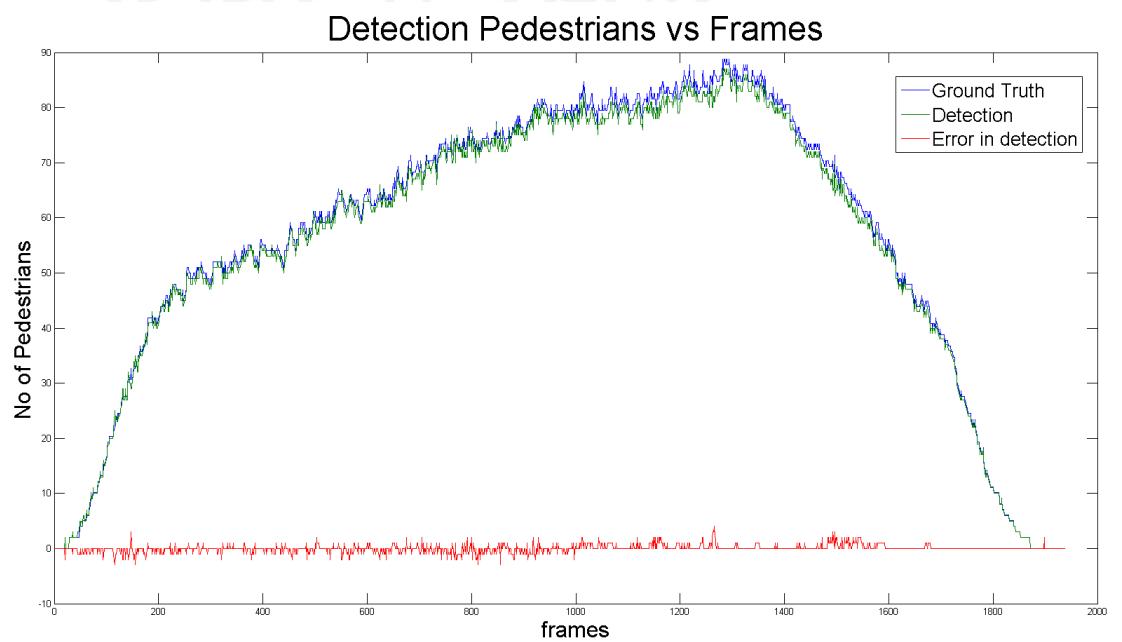


Figure 2.12: Detection Vs frame



Chapter 3

Unsupervised deep learning models for Head detection

This chapter describes the un-supervised learning models for head detection. In this chapter, instead of hand crafting features we will use representation learning algorithms to learn the representations for the images.

3.0.1 Two basic kinds of Machine Learning

Consider a machine or a learning system which is given a sequence of inputs x_1, x_2, x_3, \dots where x_t is the raw input vector at time t. This kind of input can correspond to either an image or audio or video frames. One has to distinguish between two kinds of machine learning . In supervised learning the learning system is also given a sequence of desired outputs y_1, y_2, \dots and the objective of the learning system is to predict output correctly given a raw input. The output can be a class label (in classification) or a real number as in regression problems. While in unsupervised learning the learning system simply receives inputs x_1, x_2, x_3, \dots but does not obtain the corresponding target labels from the user. It may seem difficult what a machine can learn if there is no label but one can develop a framework for Machine learning based on the idea that the goal is to build representations of the inputs which can be readily used for decision making, predictions for future. Some of the classic examples of unsupervised learning are clustering and dimensionality reduction.

3.0.2 Deep learning

Deep Learning is a machine learning framework that focuses on learning deep hierarchical models of data. Deep Learning hypothesizes that in order to learn high-level representations of data a hierarchy of intermediate representations are needed. In

the vision based models case the first level of representation could be gabor like filters, the second level could be line and corner detectors, and higher level representations could be objects and concepts. The Deep Learning equivalent would be a feed-forward neural network with many hidden layers. Many in this context being 3 or more. The theory is that if the neural net is allowed to find meaningful representations on several levels it will perform better. A key motivation for deep learning is that the brain seems to operate in a 'deep' fashion, more specifically, the neocortex has a number of attributes which speak in favour of investigating deep learning.

3.0.3 Biological Motivation

A key intuition for deep learning is that the brain seems to work in a 'deep' fashion, more specifically, the neo-cortex has a number of features which speak in favour of investigating deep architectures. One of Deep Learning's most important neocortical intuitions is that the neocortex is stratified and hierarchical. Specifically it has approximately 6 layers [27] with lower layers projecting to higher layers and higher layers projecting [11],[16] back to lower layers. The ordered nature comes from the observation that generally the upper layers represents increasingly abstract concepts and are increasingly invariant to transformations such as lighting, pose, etc. The classical example is the visual pathway in which it was found that V1, taking input from the sensory cells, reacted the strongest to simple inputs modelled very well by Gabor filter [23],[8]. As information flows from V1 to the higher areas V2 and V4, the neurons become responsive to increasingly abstract features and observe increased invariances to viewpoint, lighting, etc [12],[36],[4].

3.0.4 Deep Belief networks

Deep Belief Networks (DBNs) consists of a number of layers of Restricted Boltzmann Machines (RBMs) which are trained in a greedy layer wise fashion. A RBM is an generative undirected graphical model.

The lower layer 'x', is defined as the visible layer, and the top layer 'h' as the hidden layer. The visible and hidden layer units x and h are stochastic binary variables. The weights between the visible layer and the hidden layer are un-directed and are denoted W . In addition each neuron has a bias. The model defines the probability distribution

$$E(x, h) = - \sum_{ij} w_{ij} x_i h_j - \sum_i d_i x_i - \sum_j b_j h_j$$

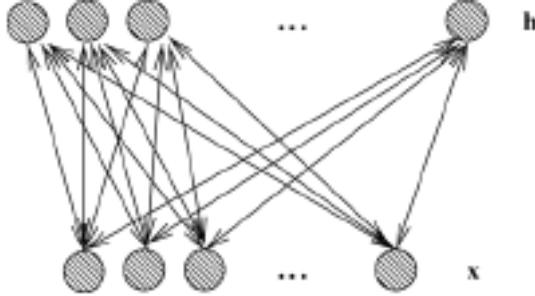


Figure 3.1: Restricted Boltzmann machine

$$p(x, h) = e^{-E(x, h)} / Z$$

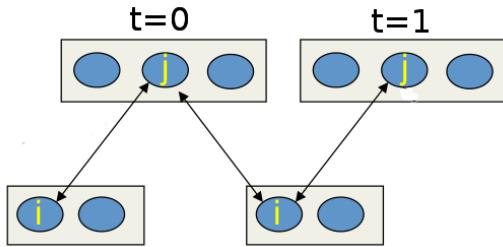
- Normalizer $Z = \sum_{x,h} e^{-E(x,h)}$ is called the partition function
- Low energy states (x, h) have high probability

Where d and b are the biases of the visible layer and the hidden layer respectively. The sum over ‘x’; ‘h’ represents all possible states of the model.

Producing a state from $p(x, h)$ is difficult.

But sampling from the marginal distributions $p(h|x)$ and $p(x|h)$ is easy. We will address the question if we are given x , the input state how to find h ?

$$\begin{aligned}
 p(h|x) &= \frac{p(x, h)}{\sum_h p(x, h)} \\
 \sum_h p(x, h) &= p(x) \\
 &= \frac{e^{-E(x, h)} / Z}{\sum_h e^{-E(x, h)} / Z} \\
 &= \frac{e^{(\sum_{ij} w_{ij} x_i h_j + \sum_j b_j h_j)} \cdot e^{(\sum_i d_i x_i)}}{\sum_{h_1 \in \{0,1\}} \sum_{h_2 \in \{0,1\}} \dots \sum_{h_J} e^{(\sum_{ij} w_{ij} x_i h_j + \sum_j b_j h_j)} \cdot e^{(\sum_i d_i x_i)}} \\
 &= \frac{\prod_j e^{(\sum_i w_{ij} x_i h_j + b_j h_j)}}{\prod_j \sum_{h_j \in \{0,1\}} e^{(\sum_i w_{ij} x_i h_j + b_j h_j)}} \\
 &= \prod_j \frac{e^{(\sum_{ij} w_{ij} x_i h_j + \sum_j b_j h_j)}}{\sum_{h_j \in \{0,1\}} e^{(\sum_{ij} w_{ij} x_i h_j + \sum_j b_j h_j)}} \\
 &= \prod_j p(h_j | x) p(h_j | x) =
 \end{aligned}$$



$$\begin{aligned}
 p(h_j = 1|x) &= \frac{e^{(b_j + \sum_i w_{ij}x_i)}}{1 + e^{(b_j + \sum_i w_{ij}x_i)}} \\
 &= \sigma(b_j + \sum_i w_{ij}x_i) \\
 p(h_j = 0|x) &= 1 - p(h_j = 1|x)
 \end{aligned}$$

By symmetry,

$$\begin{aligned}
 p(x_i = 1|h) &= \sigma(d_i + \sum_j w_{ij}h_j) \\
 p(x_i = 0|h) &= 1 - p(x_i = 1|h)
 \end{aligned}$$

The conditional probability of one layer, given the other is

$$p(h/x) = \prod_j p(h_j|x)$$

RBM's: Contrastive Divergence

RBM's are trained by Contrastive Divergence algorithm described below:

for all m

- a. Sample h_j for all j , with fixed $x = x^{(m)}$
- b. Sample x_i for all i , with fixed h
- c. Sample h_j for all j , with fixed x

$$\langle x_i h_j \rangle_{\text{data}} = \sum_m x_i^{(m)} h_j \quad \text{at } t = 0$$

$$\langle x_i h_j \rangle_{\text{model}} = \sum_m x_i h_j \quad \text{at } t = 1$$

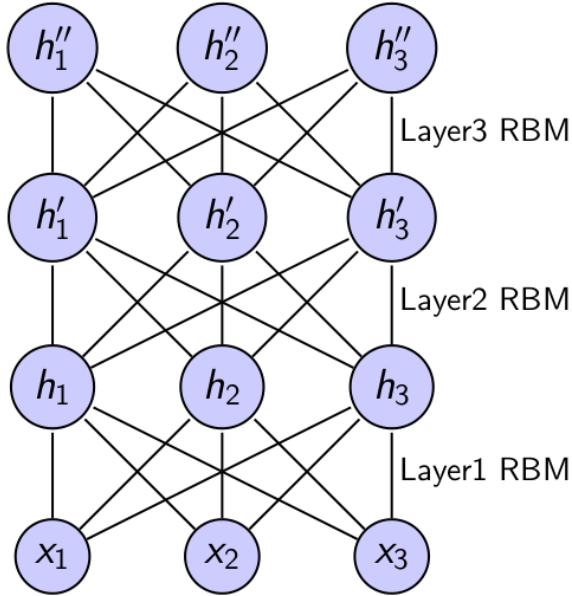


Figure 3.2: Deep Belief Network

3.0.5 Gradient of the Log-likelihood

Training RBM's to optimize $p(x)$ Maximize the probability of data $x^{(m)}$, i.e. minimize their energy.

$$\Delta w_{ij} = \eta \frac{\partial}{\partial w_{ij}} \log p(x = x^{(m)})$$

In other words, $\log p(x)$ is (-Energy).

$$\begin{aligned} & \frac{\partial}{\partial w_{ij}} \log p(x = x^{(m)}) \\ &= \frac{\partial}{\partial w_{ij}} \log \sum_h p(x = x^{(m)}, h) \\ &= \frac{\partial}{\partial w_{ij}} \log \sum_h \frac{e^{-E(x^{(m)}, h)}}{\sum_{x,h} e^{-E(x, h)}} \\ &= \frac{\partial}{\partial w_{ij}} \log \sum_h e^{-E(x^{(m)}, h)} - \frac{\partial}{\partial w_{ij}} \log \sum_{x,h} e^{-E(x, h)} \\ &= \frac{1}{\sum_h e^{-E(x^{(m)}, h)}} \sum_h \frac{\partial}{\partial w_{ij}} e^{-E(x^{(m)}, h)} - \frac{1}{\sum_{x,h} e^{-E(x, h)}} \sum_{x,h} \frac{\partial}{\partial w_{ij}} e^{-E(x, h)} \end{aligned}$$

Deep Belief Nets (DBN) are Stacked RBM's

- These networks are trained layer-by-layer (RBM's one by one), from bottom to top
- Stacked RBM's can be used to initialize Deep Neural Networks.

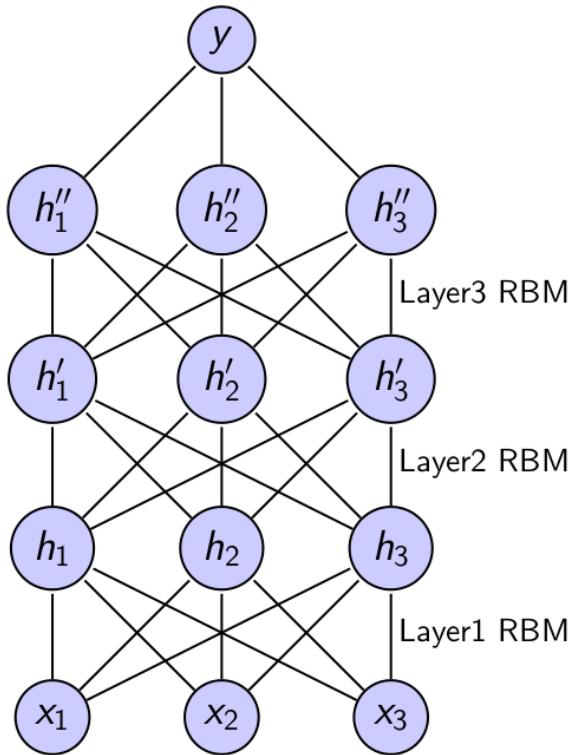


Figure 3.3: Deep neural network (DBN+output layer)

For Discriminative tasks, an output layer is added to the deep belief network to get a deep neural network.

- **Generative Pre-Training:**

In this step a layer-by-layer training of stacked RBM's using unlabeled data is done to find good initial weights.

- **Discriminative Fine-Tuning:**

Back-propagation algorithm is then used using labeled data to locally optimize the weights for better discrimination.

Weight update law:

$$\Delta w_{ij} = \eta(\langle x_i h_j \rangle_{\text{data}} - \langle x_i h_j \rangle_{\text{model}})$$

3.0.6 Experiments

Input data are the images of regions of heads and no-heads sampled from figure 3.4. The input dataset consists of a total of 4251 number of 25×25 images broken into 3023 images of class 1(heads) and rest 1228 of class 2 (no head). We have over-sampled the images from class 2 to deal with the problem of imbalanced classes.



Figure 3.4: A group of people

Then the over-sampled data has been used for further processing by the Deep Belief Network and a neural classifier.

A single layer DBN of 100 neurons was constructed. The network consisted of a single RBM of 100 hidden units. All weights and biases were initialized to be zero. Each RBM was trained on the full training set, using mini-batches of size 10, with a fixed learning rate of 0.01 for 100 epochs. One epoch is one full sweep of the data. The mini-batches were randomly selected each epoch. Having trained the first RBM the entire training dataset was transformed through the first RBM resulting in a new $N_{train} \times 100$ dataset. Having pre-trained each RBM the weights and biases were used to initialize a feed-forward neural net with 2 layers of sizes 100-10, the last 10 neurons being the output label units. The FFNN was trained using mini-batches of size 2 for 50 epochs using a fixed learning rate of 0.1 and a small L2 weight-decay of 0.00001 using back-propagation. To evaluate the performance the test set was feed-forwarded and the maximum output unit was chosen as the label for each sample resulting in an error rate of 0 %. The code run for 2 minutes.

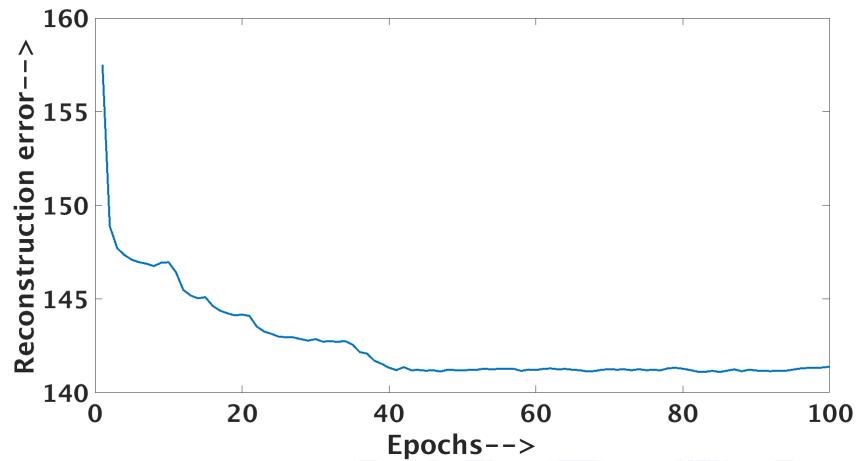


Figure 3.5: Reconstruction error

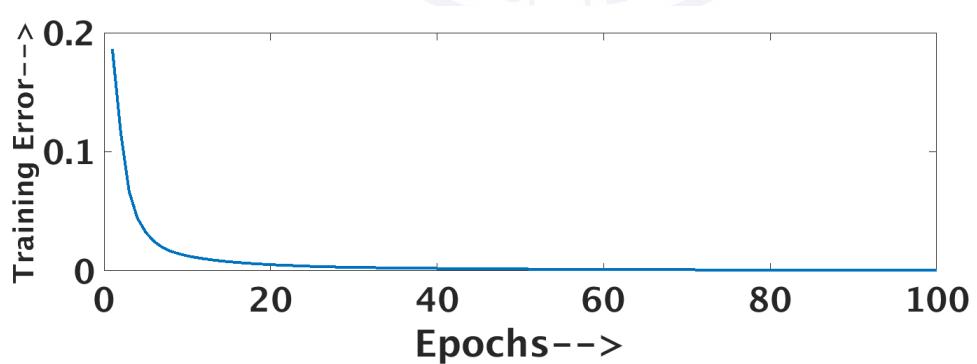


Figure 3.6: Training error

Chapter 4

Crowd Modeling

4.1 Crowd Behavior

Crowd behavior analysis in computer vision is a new area of interest in the research community which could potentially lend itself to a host of new application domains, such as automatic detection of riots or chaotic acts in crowds and localization of the abnormal regions in scenes for high resolution analysis.

To describe the individual and group behaviors in crowded scenes. Many pedestrian models have been developed over the years and have been published. These are basically grouped into two main approaches: microscopic and macroscopic as described in [26].

4.1.1 Microscopic Approach :

Microscopic models describe the space-time behavior of individual pedestrians. It describes pedestrian behavior microscopically by social fields which include social behavior of individuals.

This approach is typically to analyze not more than hundreds of pedestrians in total. In this approach, individual person behavior is simple and trajectory can be easily calculated.

The advantages of microscopic approach is more sensitive and reflects the effects of collision and interaction behavior.

4.1.2 Macroscopic Approach :

Macroscopic models focus is to describe the behavior of group of pedestrians rather than individual pedestrian. It models crowd behavior as an entity in itself and draws from gas and fluid flow approach. In this approach, the individual person in a given video data has negligible size and covers an area less than a region of 25×25 pixels.

In the following section, a survey of various self organization phenomena for crowd modelling is given, where in Lane, Stripe, Trail, Bottleneck (uni-directional, bi-directional), Stop-and-Go wave, Crowd Turbulence are described while taking example of a video of Haj pilgrims .

4.1.3 Lane Formation

Generally if the pedestrians are a few, then there is a less chance of collision with the neighborhood pedestrians. However, if the pedestrians are more and especially when there is a variation of desired directions of walking, the avoidance manoeuvres become more complex. Let us start with the special case when there are two streams of oppositely moving pedestrians, intersecting at a 180-degrees angle, i.e, bi-directional movement. From an empirical observation of such a situation, it has been reported that the pedestrians self organize into lanes of uniform direction of motion. This organization of lanes is beneficial for the pedestrians, since it reduces the frequency and strength of avoidance manoeuvres. There is nothing that indicates the organization into the lanes, follows from an individual or collective planning. For a majority of the pedestrians, it is doubtful if they are even aware of the collective pattern that they are part of. As it turns out, a very simple model where pedestrians are pushed aside from oppositely moving pedestrians can reproduce the lane formation phenomenon.

4.1.4 Stripe Formation

If we generalize the principle of lane formation to two pedestrian flows intersecting at an arbitrary angle, different from 180 degrees, it turns out that self-organization occurs also in this case, increasing the efficiency of walking for all pedestrians. This case calls for more complex avoidance manoeuvres than for bi directional streams, since the pedestrians do not only have to avoid collisions with oppositely moving pedestrians, but they eventually also have to traverse the stream of oppositely moving pedestrians. It turns out that the resulting dynamics of the pedestrians reminds of the zipper strategy observed in vehicular traffic when two lanes are merging into one. Pedestrians with the same desired walking direction turn out to form groups,

and the walkway is dynamically subdivided into stripes made up by groups with the two different directions of motion. This phenomenon is called stripe formation.

4.1.5 Pedestrian Trail Formation

For movement in terrain like grass or snow it is easier to follow the steps of others, rather than producing new paths. This behavior is similar to the pheromone-based trail following of ants but for pedestrians the role of attractive chemicals is replaced by the greater comfort of a trail that is used more often. However if the path of a predecessor deviates too much from the desired path of the pedestrian, he or she will generate a new path in the terrain. When these two principles are put together with some stochastic addend, permanent paths will eventually evolve in the terrain. Interestingly, the shape of the trails looks differently from conventionally engineered pedestrian walkways. Especially, when it comes to the angle of two paths merging or separating, naturally evolving paths will usually not have the 90-degrees angle of a T shaped intersection. Pedestrians rather prefer Y shape inspection design, which is common for freepath.

4.1.6 Bottleneck

Naturally, the most problematic locations during evacuations are bottlenecks, i.e, the most narrow places. The traditional view of a bottleneck is that the flow of pedestrians is proportional to the walkable width and that the flow characteristics are not further changed. This is roughly true for large widths, but for small bottlenecks, like doors, things become more complex.

In situations when there is a jam in front of a narrow bottleneck, e.g. jamming in front of exits during evacuations, in coordination leads to intermittent flows. These intermittent flows are characterized by an alternation between bursts of smooth out flow and times when nobody passes through the door. Intermittent flows are a big concern during evacuations, since they give very unpredictable out flows. During evacuations, people want to leave the building as fast as possible, making them more impatient and uncoordinated, which gives rise to the "*faster is slower effect*" [20], i.e. an overall decrease in the out flow as a result of a stronger urge to leave the building. This is sometimes referred to as 'escape panics' and was experimentally demonstrated in [33]. In his experiment, a number of cones were put in a bottle where water was pouring into the bottle from the bottom. A string was attached to each of the cones and the ends of the strings were given to the participants. A participant got rewarded if his/her cone came out dry, and got increasingly fined when there was water in the cone.

4.1.7 Directional Bottleneck Flows

So far only uni-directional flows through bottlenecks have been discussed, but for the case of bi-directional flows another interesting self-organization phenomenon occurs. When people are using a bottleneck in both directions and the bottleneck is narrow enough to allow only one person to pass at a time, it is obvious that the flow can either be zero or in one of the two directions. But for wider bottlenecks, we could have bi-directional flows. However, for crowded scenarios this happens only for wide bottlenecks, and not immediately when the bottleneck is wide enough to let two persons pass each other.

Once a flow is established in one of the two directions, this will generate a forward momentum that sustains the flow for some time. However, the crowd that wants to pass in the other direction will increase in number and they will eventually be eager enough to stop the flow, and change the flow direction, so they can pass the bottleneck. As this scheme is repeated, the resulting flow is oscillatory in time.

4.1.8 Stop-and-Go Waves

Until now, we have seen that, when the crowd is increasing, pedestrians are coordinating themselves with the people around, which results in self-organization.

When the density is so high that coordination is difficult a question arises as to whether self-organization survives or will it break down as well ?.

We have discovered that, when the density is so high that the level of coordination breaks down, the smooth flow turns into stop-and-go flow [21]. For stop-and-go flow, rather than everybody moving at the same speed, the motion is characterized by an alternation of moving and stopping. Basically, the pedestrians are stopped until free space appears in front of them and they can then take one or several steps forward. Looking at the phenomenon from a macroscopic level, waves of free space are propagating upstream in the crowd, anti-parallel to the movement of the pedestrians.

By analyzing video-recordings where stop-and-go waves have been observed in reality, it turns out that stop-and-go waves move with an almost constant propagation speed [25].

4.1.9 Crowd Turbulence

After the occurrence of the stop-and-go waves and the related breakdown of the pedestrian flow, the density would reach even higher values and sudden transition from stop-and-go waves to irregular flows starts to take place. These irregular flows were characterized by random, unintended displacements into all possible directions which pushes people around and this is referred to as “crowd turbulence”.

In the video, this crowd turbulence caused some individuals to stumble. As the people behind were moved by the crowd as well and could not stop, the fallen individuals were trampled, if they did not get back on their feet quickly enough. Tragically, the area of trampled people grew more and more in the course of time, as the pedestrian became obstacles for others which lead to biggest crowd disasters.

During high crowd densities, the mechanical pressure onto the people in the crowd is high, but it is not homogeneous. This conclusion can be drawn from the considerable spatial variation of densities, even for very large average densities. From this in-homogeneity of forces, eruptive events of pressure release may follow. During these events, people experience large (involuntary) displacements, which cause an irregular crowd motion into all possible directions.

As we do not observe large eddies, the similarity with fluid turbulence is limited, but there is an analogy to the turbulence at currency exchange markets.

The reason for calling this phenomenon *turbulence* is that the motion visually reminds of turbulence. Moreover, by measuring certain quantities one can show that crowd turbulence exhibits properties that are similar to those of turbulent flows.

For example, the structure function [10] $D(\Delta r) = [v(r, t) - v(r + \Delta r, t)]^2$ as a function of the displacement Δr has turbulent properties

Moreover, when measuring the velocity change $\nabla v = v(t + dt) - v(t)$ over a time period τ at a given location, and averaging over different locations ‘r’ and times ‘t’. The probability-density functions are calculated for different values of τ both in the laminar-flow regime as well as in the turbulent regime. For small values of dt in the turbulent regime, the velocity probability-density function is sharply peaked rather than Gaussian, which is typical for turbulence.

4.2 Crowd Model

As we studied the different behaviors such as self-organizing and crowd turbulence now let us create a mathematical model of the crowd based on [26]. Let us assume that pedestrians are moving around in a region covering A_r (m^2). For a given time ' t ' there are N_r number of pedestrians and the identification of the i^{th} pedestrian yields a location $\vec{X}_i(t)$ which can be given as

$$X_i(t) = [x_i \ y_i]$$

By comparing the location in previous frame, we can also determine velocity of individual person as \vec{v}_i given in figure 4.1.

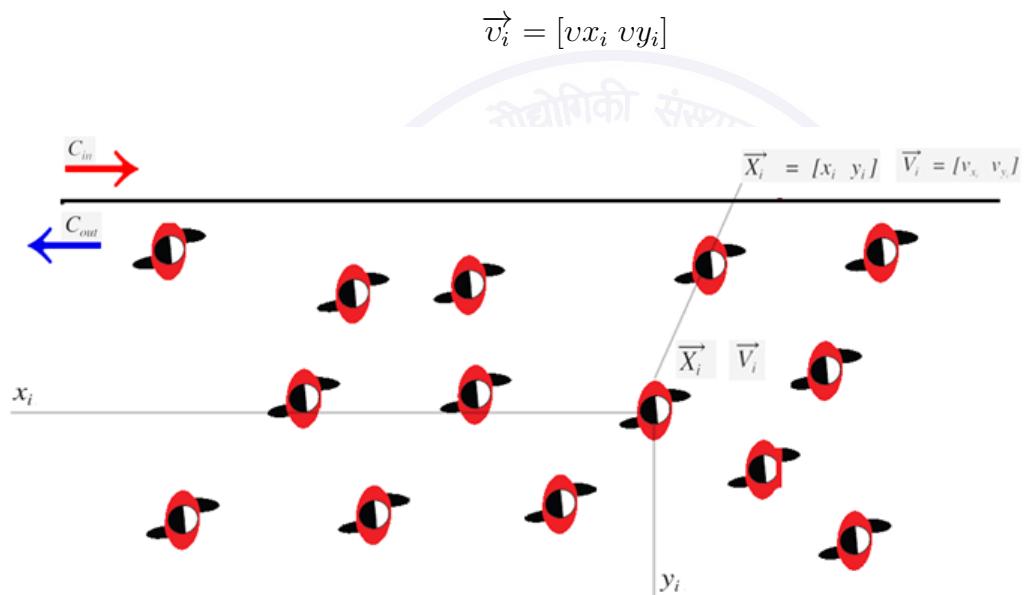


Figure 4.1: Crowd Model

4.2.1 Parameters of Crowd Model

The behavior of crowd is most complex how ever we are trying to explain some of the crowd behavior by parameters such as Local Density ρ , Local Velocity \vec{v} , Global Density ϱ , Variation in Velocity $\nabla \vec{v}$, Pressure P . In principle, the Global Density $\varrho(t)$ is calculated by Local Density ρ and Variance in Velocity $\nabla \vec{v}$ is calculated by Local velocity \vec{v} and Pressure depends both on local density and velocity variance.

The global density is measured by counting all pedestrians(N_r) in given area A_r as:

$$\varrho = \frac{N_r}{A_r} \quad (4.1)$$

4.2.2 View Point Variation of Video Dataset

In order to verify or to implement the crowd model, empirical evidence must eventually, that supports the theories. For empirically evaluating the crowd model we need geographical location $X_i(t)$ of the pedestrian. The geographical location $X_i(t)$ can be calculated by different ways like based on video, infra-red sensors, laser device or manually evaluating the location. Video based evaluation is the simplest way of evaluating $X_i(t)$, however there is a scaling constant k that should be considered which is calculated as no.of pixels covering 1 meter geographically.

Before implementing the model, let us see different types of video data where we can implement the model. Crowd Video Dataset can be classified into three types different view points **Different angles of camera**

- 1. Perpendicular View** In this type of dataset the camera is placed perpendicular to the ground where crowd gathering is going to take place. In this type of view the size of head is same in entire Video dataset and there is no chance of pedestrian being occluded by the other pedestrians. In other words, the pedestrian can't hide from the camera as shown in figure 4.2.



Figure 4.2: Perpendicular View of Camera

- 2. Perspective View** In this type of dataset, the camera is placed faraway from crowd and angle is not perpendicular to the ground. In these types of view, the size of head is not same in the entire video dataset, the person who are near camera has large size. As the angle of camera increases, the chances of

occlusion also increases and at a certain angle the pedestrian is occluded for such type of video dataset. We can't implement our model on the view as shown in figure 4.3.



Figure 4.3: Perspective View of Camera

3. **High Altitude View** In this type of dataset, the camera is placed at a very large height (an aerial view of the crowd). If an individual person in the video is very small, say almost 4-5 pixels. In this type of data finding an individual manually is very difficult as shown in figure 4.4.

From above information, we can tell that we can't implement the crowd model on all types of dataset especially on perspective view because at a certain angle the person would be occluded and there will be large variation of head which makes it difficult to find the head location.

4.2.3 Local Density

Traditional definition of crowd density is given as no. of person's in 1 meter square normally. But this is only valid for a uniform pedestrian distribution in the area.



Figure 4.4: High Altitude View of Camera

But if the distribution of pedestrians is not uniform in the area, then the value of density is inverse of the area of each pedestrian as shown in figure 4.5, there is also another type of Local density distribution by Helbing [26] similar to that of the Gaussian filter as shown in figure 4.6.

Vornoic Distribution based Local Density :(ρ_V)

In Vornoic Distribution based Local Density(ρ_V) the location of pedestrian X_i is considered as a nucleus and the entire region(A_r) is divided into individual regions of vornoic cells V_i of area a_i . as shown in figure 4.5. The smaller regions indicate high density and larger regions indicate low density [49]. Based on the area of individual pedestrian, we can determine the local density and it is given as:

$$\rho(V_i) = \frac{1}{a_i} \quad (4.2)$$

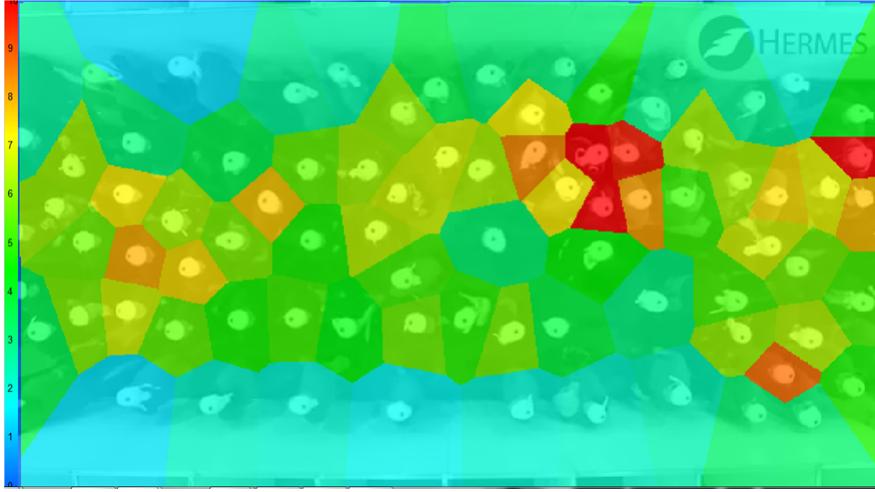


Figure 4.5: Vornoic Distribution Based Local Density

We can calculate the average of the local density

$$\begin{aligned}
 avg(\rho_V) &= \frac{\iint_{A_r} \rho_V \, dx \, dy}{\iint_{A_r} \, dx \, dy} \\
 &= \frac{\sum_j \iint_{a_j} \rho_V(V_j) \, dx \, dy}{A_r} \\
 &= \frac{\sum_j \iint_{a_j} \frac{1}{a_i} \, dx \, dy}{A_r} \\
 &= \frac{N_r}{A_r} = \varrho
 \end{aligned} \tag{4.3}$$

From above derivation, we can find the global density ϱ .

$$\varrho = avg(\rho_V)) \tag{4.4}$$

We also verified the above statement. We have taken ground truth locations of pedestrians \vec{X} from a video dataset and calculated the voroic local density. We have observed that global density and average of Vornoic local density are same in all the frames as shown in figure 4.8.

Gaussian Distribution based Local Density (ρ_G)

Centered at the location \vec{X}_i of i^{th} pedestrian we assume the density is gaussian distributed of having a peak of 1 person per $meter^2$ and having standard deviation σ . Local density can be given by $\rho_G(\vec{X})$ at a location \vec{X} via the formula

$$\rho_G(\vec{X}) = \sum_j \frac{e^{-\frac{-||\vec{X} - \vec{X}_i||^2}{2\sigma^2}}}{\iint_{A_r} e^{-\frac{-||\vec{X}||^2}{2\sigma^2}}} = \frac{1}{\sqrt{2\pi}\sigma} \sum_j e^{-\frac{-||\vec{X} - \vec{X}_i||^2}{2\sigma^2}} \tag{4.5}$$

where, standard deviation σ is an arbitrary value

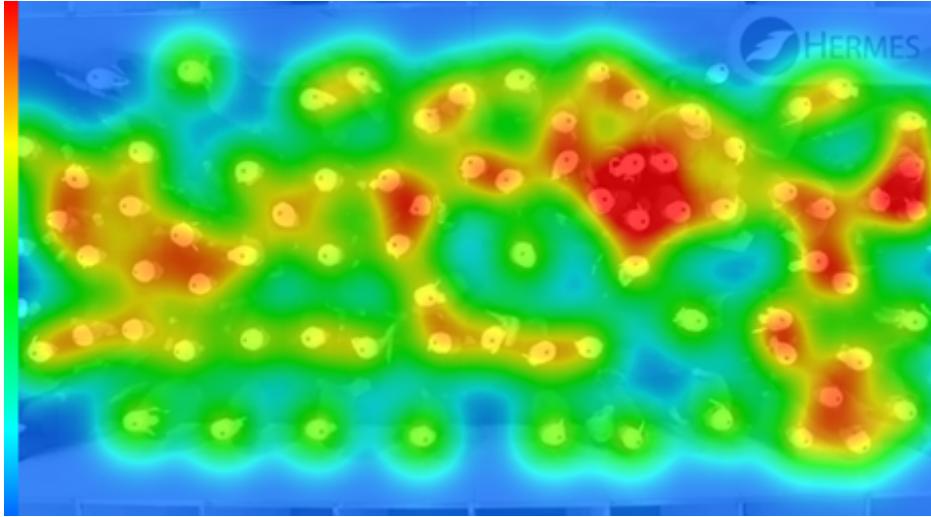


Figure 4.6: Gaussian Distribution based Local Density

$$\varrho = \frac{N_r}{A_r} \quad (4.6)$$

The global velocity is determined by calculating the average velocity of all persons in that area.

$$V = \sum_j \frac{v_j}{N_r} \quad (4.7)$$

And the global flow is obtained as global density times the global velocity (the number of people passing a long cross section per unit time, divided by its length.)

$$Q = \varrho \cdot V = N_r / \text{time} \quad (4.8)$$

The global density $\varrho(t)$ for different values of σ evaluated over average of local density ρ can be given as:

$$\begin{aligned} \text{avg}(\rho_G) &= \frac{\iint_{A_r} \rho_G \, dx \, dy}{\iint_{A_r} \, dx \, dy} \\ &= \frac{1}{\sqrt{2\pi}\sigma} \frac{\sum_j \iint_{A_r} e^{-\frac{||\vec{x} - \vec{x}_j||^2}{2\sigma^2}} \, dx \, dy}{A_r} \\ &= \frac{1}{\sqrt{2\pi}\sigma} \frac{\sum_j \sqrt{2\pi}\sigma}{A_r} \\ &= \frac{N_r}{A_r} \\ &= \varrho \end{aligned} \quad (4.9)$$

From above derivation, one can find the global density by:

$$\varrho = \text{avg}(\rho_G) \quad (4.10)$$

From above derivation, we can tell that the global density ϱ is independent of σ . So we empirically evaluated the global density of Ground Truth location in a dataset and the average(ρ) for different values of σ .

The **Error in Global Density** is calculated as the rms error in $\text{avg}(\rho)$ given as:

$$\text{Local Error}(\sigma) = \sqrt{\frac{1}{F} \sum_{f=1}^F [\varrho(f) - \text{avg}(\rho_G(\sigma, f))]^2} \quad (4.11)$$

where ‘f’ is current frame and ‘F’ is total no of frames. We observed that the Global

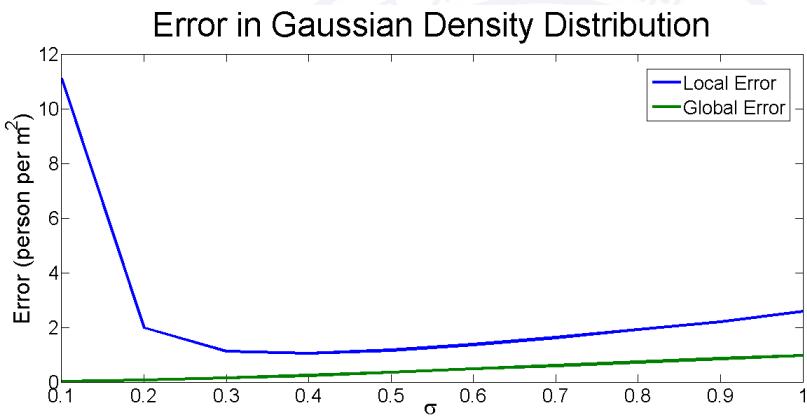


Figure 4.7: Error in Global Density ϱ_G , Local Density ρ_G Vs σ

Error is small (< 1 person per m^2) and error increases with increasing σ as shown in figure 4.7. Thus we empirically evaluated that the Global Density is average of the Gaussian Local Density ρ_G . In order to find desired sigma value which has similar distribution as Vornoic Local Density ρ_V we evaluated the rms error in local density of gaussian ρ_G for all σ as shown in figure 4.7 and the error is given by

$$\text{Local Error}(\sigma) = \sqrt{\frac{1}{F} \sum_{f=1}^F [\text{avg}(\rho_V(f)) - \rho_G(\sigma, f)]^2} \quad (4.12)$$

We observed that as sigma increases the error is reducing till $\sigma = 0.4$ having density error of Local Error = 1.047 (person/m^2) further increasing sigma caused increase in error. So we have used the sigma $\sigma = 0.4$ in our Gaussian Local Density.

4.2.4 Results of Density

In order to find the local density ρ of the given image we need to find the head location $X_i(t)$ which is calculated by using HOG features 2.1 and results of global density for all frames is given in figure 4.8.

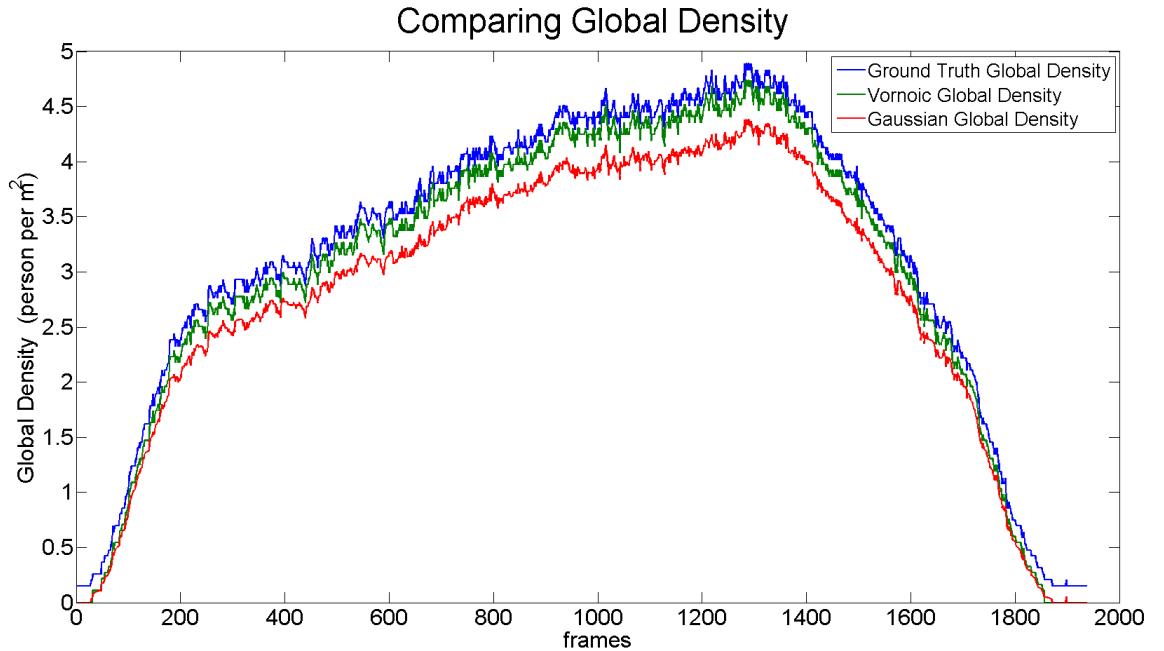


Figure 4.8: Global Density Vs frame

4.2.5 Local Velocity

Velocity Distribution of the region(A_r) can be considered to be either a gaussian or a vornoic distribution of the individual pedestrian X_i . For example, in the case of Microscopic Approach referred in section 4.1.2 where number of pedestrians is not more than few hundred, we can use tracking algorithm like kalman filter tracking [19] and results of kalman tracking is given in figure 4.9 to find the velocity of individuals so that we find the velocity distribution of entire frame.

4.2.6 Results of Velocity by kalman tracking

Vornoic Distribution based Local Velocity

The definition of Vornoic Velocity is similar to Vornoic Density. In Vornoic Distribution based Local Velocity, the location of pedestrian X_i is considered as a nucleus and the entire region (A_r) is divided into individual region of vornoic cells V_i having velocity v_i [49]. Based on the area of individual pedestrian, we can determine the

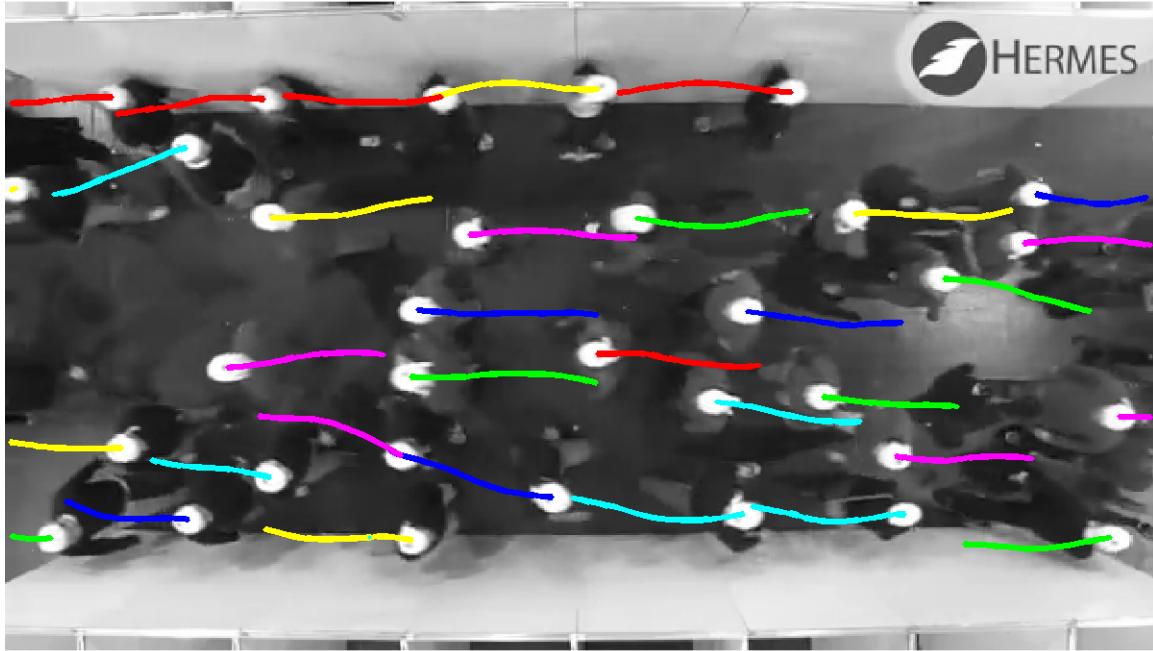


Figure 4.9: Kalman Tracking

Local Velocity and it is given as

$$v(V_i) = v_i \quad (4.13)$$

Gaussian Distribution based Local Velocity

The local velocities can also be considered as Gaussian distribution and given as

$$v(\vec{X}) = \sum_j \vec{v}_j \cdot e^{-\frac{\|\vec{X} - \vec{X}_j\|^2}{2\sigma^2}} \quad (4.14)$$

and local speeds as $v(\vec{X}) = |v(\vec{X})|^2$

Optical flow based Local Velocity estimation

The above velocity distributions (Gaussian, Voronoi) is valid if we are able to track the individual persons, but however in Macroscopic approach referred in section 4.1.1 in such a case, the number of pedestrians are very large (in thousands) and the area occupied by a pedestrian is very small(few pixels), hence detection of individual person is very difficult and applying tracking algorithm will be much more complex

and non-approachable but we can approximate velocity of individual person as the optical flow for the frame as shown in figure 4.10.

However this is only valid if the pedestrian occupies very small area(few pixels) but if the pedestrian occupied area is large, this approach gives unwanted velocity and further we can't find the change in velocity by optical flow for large pedestrian size.

4.2.7 Variation in Velocity

The Change in the velocity is one of key factors for crowd dynamics [26]. Although change in velocity does not give any information about the number of pedestrians or crowd density but still it gives the information of how the pedestrians are moving. The individual pedestrian behavior and velocity most importantly depends upon the surrounding behavior and velocity of pedestrians rather than the previous velocity of the individual pedestrians

Here ∇v_i represents change but with respect to neighborhood pedestrians which means the velocity of individual changes with respect to the velocity of the surrounding pedestrians.

For Macroscopic Approach 4.1.2: we can consider the above parameters (pressure, velocity, density) to be continuous in space \vec{X} which means at each pixel considered, an individual person has his/her own velocity v . Then variation in velocity \vec{v} is given as

$$\nabla v = v - \text{avg}(v) \quad (4.15)$$

The above equation $\text{avg}(v)$ is the average velocity in the neighborhood region at that person location. Since in macroscopic approach, each pixel is a person so use Gaussian average as the $\text{avg}(v)$ of person and given by

$$\nabla v = v - v * N(0, \sigma^2) \quad (4.16)$$

here

$$\text{avg}(v) = v * N(0, \sigma^2) \quad (4.17)$$

Here N is a Normal Distribution Function with mean zero and standard deviation σ . The sigma is arbitrarily chosen. We use $\sigma = 0.4$ as we estimated in section 4.2.3.

Note that the above equation for $\text{avg}(v)$ is valid only when we assume the adjacent pixel is also a pedestrian having velocity value and v is calculated either by

individual velocity or Optical flow based Local Velocity.

For microscopic approach 4.1.1: the distribution is mostly discrete, so for the variation in velocity of the i^{th} pedestrian we use a similar approach as above and $\text{avg}(v)$ is given as gaussian average of neighboring pedestrians with $\sigma = 0.4$ and given as

$$\text{avg}(v_i) = \frac{\sum_j (v_j) \cdot e^{\frac{-\|x_j - x_i\|^2}{2\sigma^2}}}{\sum_j e^{\frac{-\|x_j - x_i\|^2}{2\sigma^2}}} \quad (4.18)$$

and the Variation in the velocity of i^{th} is given as

$$\nabla v_i = v_i - \text{avg}(v_i) \quad (4.19)$$

4.2.8 Pressure

In order to measure the irregularities in crowd which is also referred as crowd turbulence, we use information about density and change in velocity [26]. The Crowd Pressure is given as:

$$P(\vec{X}_i) = \rho_i \nabla \vec{v}_i \quad (4.20)$$

The above equation is similar to the Newtonian Pressure where we consider area of the individual as crowd density and mass of the individual is considered as a unit value.

$$\begin{aligned} P(\vec{X}_i) &= \frac{1}{A_i} F_i \\ P(\vec{X}_i) &= \rho_i * 1 * \nabla \vec{v}_i \\ P(\vec{X}_i) &= \rho_i \nabla \vec{v}_i \end{aligned} \quad (4.21)$$

where ∇v_i is referred as variation in velocity but not change of velocity with respect to time but with respect to \vec{X} .

4.2.9 Results of Pressure

We have evaluated the pressure parameter values and the visual results of pressure changing in a given image as shown in figure 4.10 and the pressure variation for overall frame is shown in figure 4.12. The results are similar to the results produced by [26] as shown in the plot 4.11.

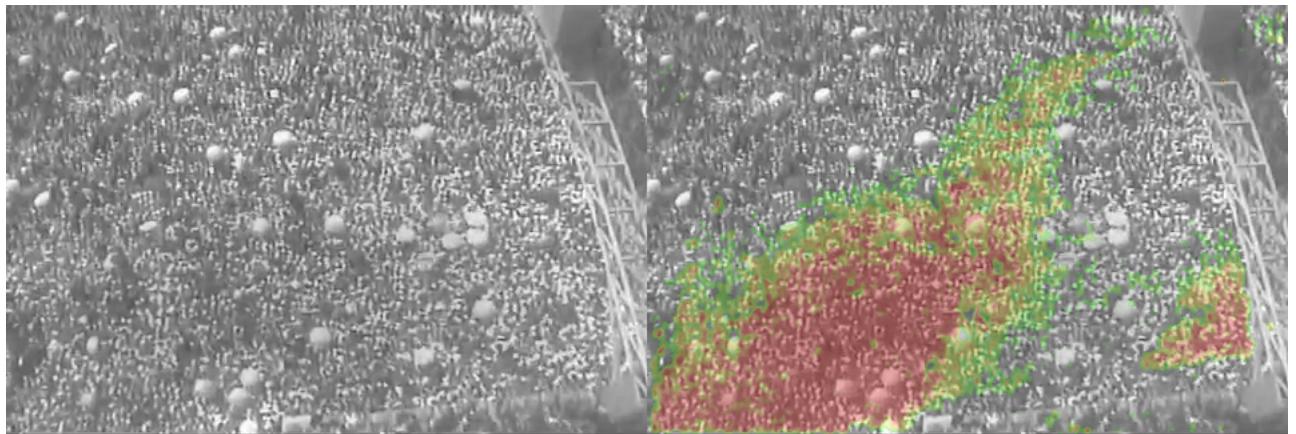


Figure 4.10: Visual Results of pressure

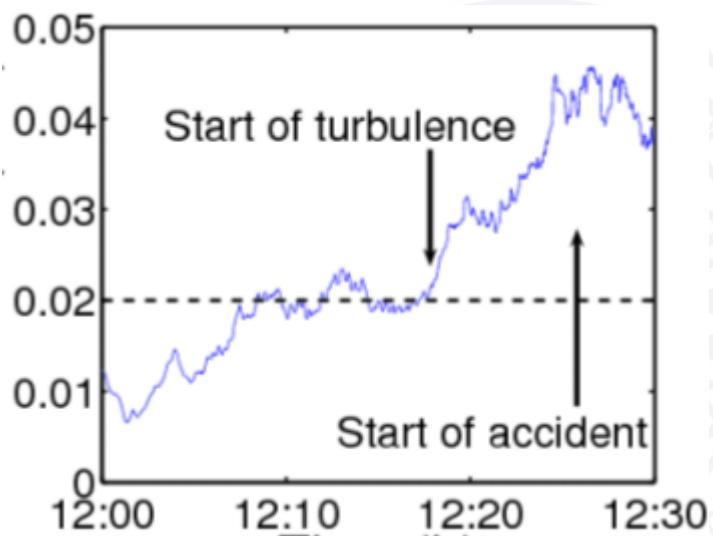


Figure 4.11: reference Pressure Vs frame

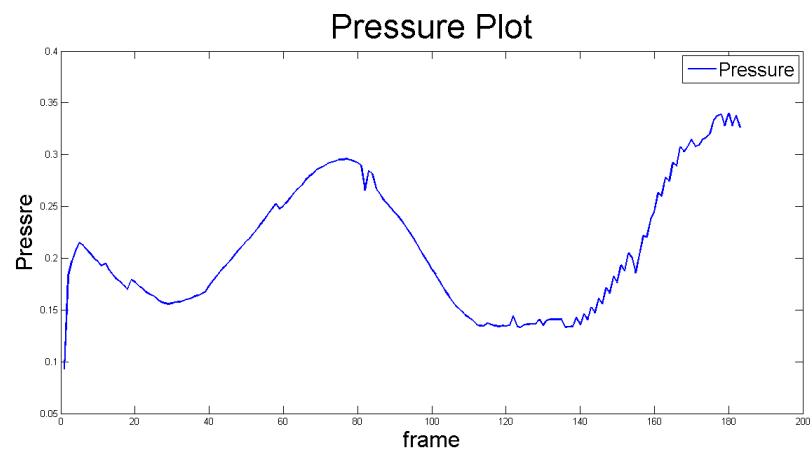


Figure 4.12: Pressure vs frame



Chapter 5

Conclusion and Future Scope

5.1 Conclusion

The main contributions of the present work are as follows:

In the 2nd chapter,

- In 2.4 we use novel Local Maxima/Local Minima method in order to estimate head in an image, where we use simple Difference in median filter in order to estimate the Head Location which gives more advantage compared to DoG and LoG in cases of detecting head even if they are overlapped, robust to noise and improves the detection rate. This method is useful for various sizes even if the size of the head is very small as shown in figure 4.4.
- The major contribution to the work is by combining hog with local maxima and minima 2.5 algorithm where it reduces the computation time, hence paving a way for implementation of the crowd model in real time.

In 3rd chapter

- Unsupervised deep learning models have been used for detecting heads and achieving close to 100% accuracy.

In 4th chapter

- We have tried to implement the crowd model for different camera angles such as perpendicular view, perspective view, high altitude view even though there are limitations in the implementation.
- We also theoretically redefine the Gaussian Local Density 4.2.3 based on Helbing et al [26], which is independent of camera angle and height of camera unlike traditional approaches [26], which is more realistic than traditional way of definition.

- We normalized the Gaussian Local Density 4.2.3 which is independent of the camera angle and height. We evaluated best sigma ($\sigma = 0.4$) value suitable for the Gaussian Local Density.
- We have redefined the variation in velocity 4.2.7 which can be used for both microscopic and macroscopic approaches.
- We have implemented a novel, simple way of calculating the variation in velocity by using optical flow 4.2.6 as velocity and the results are similar to the traditional methods of calculating variation in velocity.

5.2 Future Scope

In order to avoid crowd disaster, crowd model can be implemented in crowd scene or event used for crowd analysis and crowd monitoring. This can serve as useful input for the security and crowd management decisions regarding the times when pedestrian moment becomes unstable. This crowd modeling system can be set up and run on 24 video cameras mounted primarily on the critical area (dangerous area or more crowded area).

For basic monitoring, cameras were mounted in two layers, one covering all input entry points and all exit points of the crowd event, and another layer covering all potentially dangerous areas, for example, highly crowded area or turning points of the crowd event.

With cameras covering all the inputs and output points to and from the crowded scene, we were able to find all the flows of pedestrians and determine the difference between in flows and out flows, and from this information the increase or decrease of the crowd density inside of each area can be calculated. Further on, we could compare the pedestrian flows under each camera to the available capacity of the corresponding section. By this, the security forces were able to assess the potential risk of over-utilization of different parts of the system. This information supported their decisions to stop or redirect flows of pedestrian depending on their spatial distribution. The above modeling is based on Helbing [25] as shown in figure 5.1 5.2

5.3 Some Measures to Improve Crowd Safety

In the interest of crowd safety, densities higher than 3-4 persons per square meter in large crowds and particularly the onset of stop-and-go waves or crowd turbulence must be avoided. Therefore, a combination of the following measures are recommended:

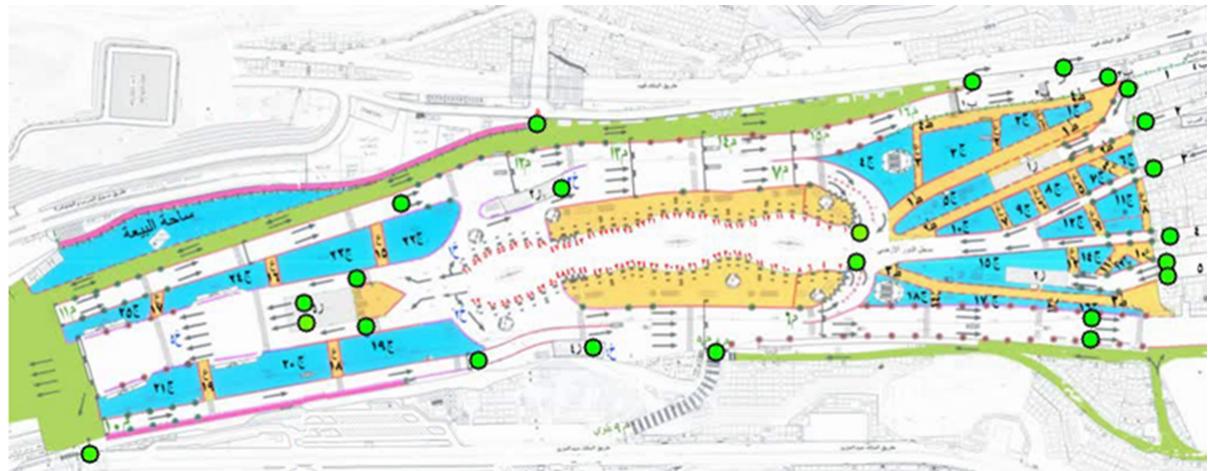


Figure 5.1: Large View

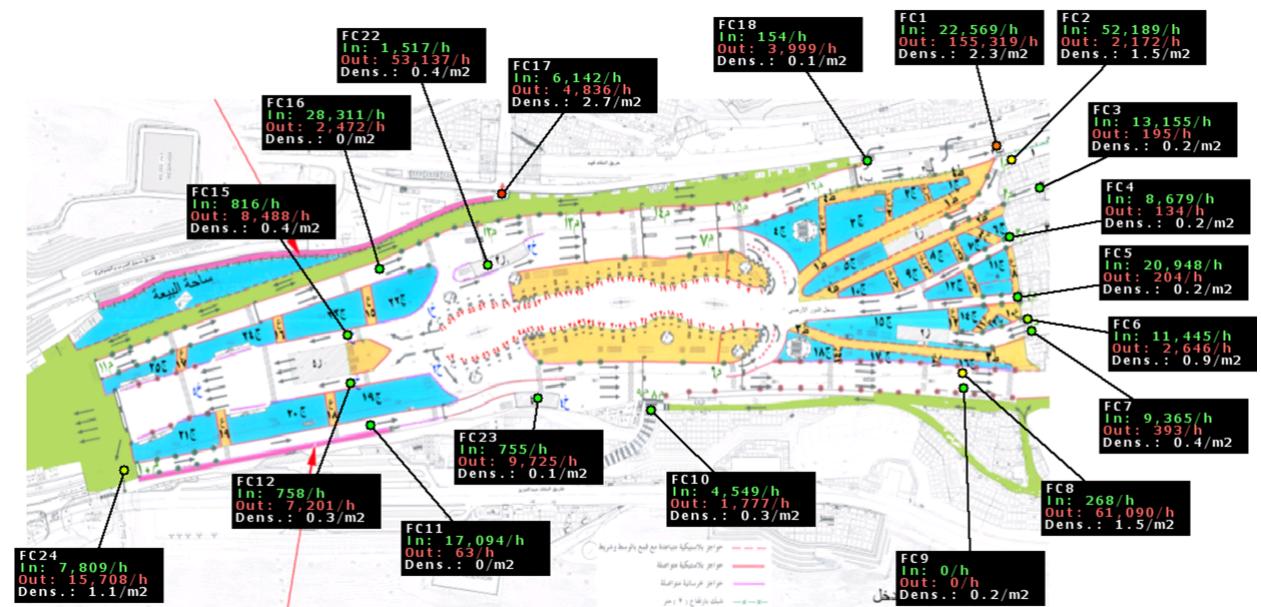


Figure 5.2: Large Scale Estimate

- Design:** The infrastructure should be designed in a way that no bottlenecks or objects (e.g. luggage) will obstruct the flow. This can be quite challenging, if the usage patterns are changing. In particular, it must be avoided that the outflow capacity is smaller than the inflow capacity. Accumulations of large crowds should be avoided. If this is impossible, the outflow capacities must be dimensioned such that the time to evacuate the area is much shorter than the time to fill it. (It should not exceed a few minutes.) Furthermore, a system of emergency routes or, more generally, a “valve system” is required to be able to reduce pressures in certain areas of the system, where needed.
- Operation:** The infrastructure should be operated in a way that avoids counter-flows or intersecting flows. Even merging flows may cause serious

problems. Flows of people should be re-balanced if there is a large utilization in certain parts of the system while there is still available capacity in others. Obstacles (including people blocking the ways) should be removed from areas intended for moving. Some ways should be reserved for emergency operation and protected from public access.

- **Monitoring:** Areas of accumulation or any possible conflict points (including crossing flows or bottlenecks, if these are really unavoidable in the system) must be monitored during highly frequented time periods. To support the job of the monitoring crew and maneuver security forces to the right places, it is helpful to display additional information in the surveillance videos, visualizing, for example, the density, the flow, and/or the crowd pressure.
- **Crowd Management:** In certain events, the flow must be suitably limited to the safe capacity of the system (which should consider a safety margin of 30%). This may be done by applying a scheduling program, which is a plan regulating the timing and routing of groups of people. The compliance with the scheduling program must be carefully monitored (e.g. by control points and/or GPS tracking), and deviations from it must be counter-acted (e.g. by fines). Moreover, an adaptive re-scheduling should be possible in order to respond properly to the actual conditions in the system. It is even more favorable to have a simulation tool for the prediction of the flows in the system. This would allow an anticipative crowd management.
- **Contingency Plans:** For situations where the system enters a critical state for whatever reason (e.g a fire, an accident, violent behavior, or bad weather conditions) one needs to have detailed contingency plans which must be worked out and exercised in advance. The above is a list of only some of the measures that can be taken to improve crowd safety.

Bibliography

- [1] Saad Ali and Mubarak Shah. A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–6. IEEE, 2007.
- [2] Saad Ali and Mubarak Shah. Floor fields for tracking in high density crowd scenes. In *Computer Vision–ECCV 2008*, pages 1–14. Springer, 2008.
- [3] Ernesto L Andrade, Scott Blunsden, and Robert B Fisher. Modelling crowd scenes for event detection. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 1, pages 175–178. IEEE, 2006.
- [4] Moshe Bar, Roger BH Tootell, Daniel L Schacter, Doug N Greve, Bruce Fischl, Janine D Mendola, Bruce R Rosen, and Anders M Dale. Cortical mechanisms specific to explicit visual object recognition. *Neuron*, 29(2):529–535, 2001.
- [5] Gabriel J Brostow and Roberto Cipolla. Unsupervised bayesian detection of independent motion in crowds. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 594–601. IEEE, 2006.
- [6] Antoni B Chan and Nuno Vasconcelos. Mixtures of dynamic textures. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 641–647. IEEE, 2005.
- [7] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.
- [8] John G Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *JOSA A*, 2(7):1160–1169, 1985.
- [9] Kalyanmoy Deb and Tushar Goel. A hybrid multi-objective evolutionary approach to engineering shape design. In *Evolutionary Multi-Criterion Optimization*, pages 385–399. Springer, 2001.

- [10] Charles R Doering and John D Gibbon. *Applied analysis of the Navier-Stokes equations*, volume 12. Cambridge University Press, 1995.
- [11] ER Ergenzerger, MM Glasier, JO Hahm, and TP Pons. Cortically induced thalamic plasticity in the primate somatosensory system. *Nature neuroscience*, 1(3):226–229, 1998.
- [12] Jack L Gallant, Charles E Connor, Subrata Rakshit, James W Lewis, and DAVID C Van Essen. Neural responses to polar, hyperbolic, and cartesian gratings in area v4 of the macaque monkey. *Journal of neurophysiology*, 76(4):2718–2739, 1996.
- [13] Tarak Gandhi and Mohan Manubhai Trivedi. Pedestrian protection systems: Issues, survey, and challenges. *Intelligent Transportation Systems, IEEE Transactions on*, 8(3):413–430, 2007.
- [14] Dariu M Gavrila. A bayesian, exemplar-based approach to hierarchical shape matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(8):1408–1421, 2007.
- [15] Dariu M Gavrila and Vasanth Philomin. Real-time object detection for smart vehicles. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 87–93. IEEE, 1999.
- [16] HT Ghashghaei, CC Hilgetag, and H Barbas. Sequence of information processing for emotions based on the anatomic dialogue between prefrontal cortex and amygdala. *Neuroimage*, 34(3):905–923, 2007.
- [17] Chunhui Gu, Jasmine J Lim, Pablo Arbeláez, and Jagannath Malik. Recognition using regions. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1030–1037. IEEE, 2009.
- [18] Jinnian Guo, Xinyu Wu, Tian Cao, Shiqi Yu, and Yangsheng Xu. Crowd density estimation via markov random field (mrf). In *Intelligent Control and Automation (WCICA), 2010 8th World Congress on*, pages 258–263. IEEE, 2010.
- [19] Andrew C Harvey. *Forecasting, structural time series models and the Kalman filter*. Cambridge university press, 1990.
- [20] Dirk Helbing, Illés Farkas, and Tamas Vicsek. Simulating dynamical features of escape panic. *Nature*, 407(6803):487–490, 2000.
- [21] Dirk Helbing, Anders Johansson, and Habib Zein Al-Abideen. Dynamics of crowd disasters: An empirical study. *Physical review E*, 75(4):046109, 2007.

- [22] Dirk Helbing and Peter Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995.
- [23] David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106–154, 1962.
- [24] Roger L Hughes. A continuum theory for the flow of pedestrians. *Transportation Research Part B: Methodological*, 36(6):507–535, 2002.
- [25] Anders Johansson. Data-driven modeling of pedestrian crowds. 2008.
- [26] Anders Johansson, Dirk Helbing, Habib Z Al-Abideen, and Salim Al-Bosta. From crowd dynamics to crowd safety: a video-based analysis. *Advances in Complex Systems*, 11(04):497–527, 2008.
- [27] Eric R Kandel, James H Schwartz, Thomas M Jessell, et al. *Principles of neural science*, volume 4. McGraw-hill New York, 2000.
- [28] G Le Bon. The crowd: A study of the popular mind. london: Ernest benn. *English edition originally published*, 1896.
- [29] Bastian Leibe, Edgar Seemann, and Bernt Schiele. Pedestrian detection in crowded scenes. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 878–885. IEEE, 2005.
- [30] Yazhou Liu, Shiguang Shan, Xilin Chen, Janne Heikkila, Wen Gao, and Matti Pietikainen. Spatial-temporal granularity-tunable gradients partition (stggp) descriptors for human detection. In *Computer Vision–ECCV 2010*, pages 327–340. Springer, 2010.
- [31] Yazhou Liu, Shiguang Shan, Wenchao Zhang, Xilin Chen, and Wen Gao. Granularity-tunable gradients partition (ggp) descriptors for human detection. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1255–1262. IEEE, 2009.
- [32] Jorge S Marques, Pedro M Jorge, Arnaldo J Abrantes, and JM Lemos. Tracking groups of pedestrians in video sequences. In *Computer Vision and Pattern Recognition Workshop, 2003. CVPRW'03. Conference on*, volume 9, pages 101–101. IEEE, 2003.
- [33] Alexander Mintz. Non-adaptive group behavior. *The Journal of abnormal and social psychology*, 46(2):150, 1951.

- [34] Karsten Peters, Anders Johansson, Audrey Dussutour, and Dirk Helbing. Analytical and numerical investigation of ant behavior under crowded conditions. *Advances in Complex Systems*, 9(04):337–352, 2006.
- [35] Fatih Porikli. Integral histogram: A fast way to extract histograms in cartesian spaces. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 829–836. IEEE, 2005.
- [36] Fangtu T Qiu and Rüdiger Von Der Heydt. Figure and ground in the visual cortex: V2 combines stereoscopic cues with gestalt rules. *Neuron*, 47(1):155–166, 2005.
- [37] Pini Reisman, Ofer Mano, Shai Avidan, and Amnon Shashua. Crowd detection in video sequences. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 66–71. IEEE, 2004.
- [38] Payam Sabzmeydani and Greg Mori. Detecting pedestrians by learning shapelet features. In *Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on*, pages 1–8. IEEE, 2007.
- [39] Edgar Seemann, Bastian Leibe, Krystian Mikolajczyk, and Bernt Schiele. An evaluation of local shape-based features for pedestrian detection. In *BMVC*, volume 5, page 10. Citeseer, 2005.
- [40] Amnon Shashua, Yoram Gdalyahu, and Gaby Hayun. Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 1–6. IEEE, 2004.
- [41] Kardi Teknomo. Microscopic pedestrian flow characteristics: Development of an image processing data collection and simulation model. *Diss. Tohoku Univ*, 2002.
- [42] Peter Tu, Thomas Sebastian, Gianfranco Doretto, Nils Krahnstoever, Jens Rittscher, and Ting Yu. Unified crowd segmentation. In *Computer Vision–ECCV 2008*, pages 691–704. Springer, 2008.
- [43] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.
- [44] Bo Wu and Ram Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 90–97. IEEE, 2005.

- [45] Bo Wu and Ram Nevatia. Cluster boosted tree classifier for multi-view, multi-pose object detection. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.
- [46] Li-Qun Xu and Arasanathan Anjulan. Crowd behaviours analysis in dynamic visual scenes of complex environment. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 9–12. IEEE, 2008.
- [47] Wenjian Yu and Anders Johansson. Modeling crowd turbulence by many-particle simulations. *Physical review E*, 76(4):046105, 2007.
- [48] Beibei Zhan, Dorothy N Monekosso, Paolo Remagnino, Sergio A Velastin, and Li-Qun Xu. Crowd analysis: a survey. *Machine Vision and Applications*, 19(5-6):345–357, 2008.
- [49] J Zhang, W Klingsch, A Schadschneider, and A Seyfried. Transitions in pedestrian fundamental diagrams of straight corridors and t-junctions. *Journal of Statistical Mechanics: Theory and Experiment*, 2011(06):P06004, 2011.
- [50] Qiang Zhu, Mei-Chen Yeh, Kwang-Ting Cheng, and Shai Avidan. Fast human detection using a cascade of histograms of oriented gradients. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1491–1498. IEEE, 2006.