OCCLUSION HANDLING IN MULTIPLE OBJECT TRACKING

by

GauravPawar,

M.Tech, signal processing and digital design,
Delhi Technological University, Delhi, 2015

SUBMITTED TO THE DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGG,
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF TECHNOLOGY IN SIGNAL PROCESSING AND DIGITAL DESIGN

AT

DELHI TECHNOLOGICAL UNIVERSITY, DELHI

JULY 2015

Under the Supervision of

MR. Rajesh Rohilla,
Associate Professor,
Department of Electronics & Communication Engineering,
Delhi Technological University,
Delhi, India.

DEPARTMENT OF ELECTRONICS AND COMMUNICATIONENGINEERING
Delhi Technological University,
(*formerly* Delhi College of Engineering)
Bawana Road, Delhi – 110042

Certificate

This is to certify that the dissertation title **"Occlusion Handling in Multiple Object Tracking"**submitted by **Mr GauravPawar**, Roll. No. 2K13/SPD/06, in partial fulfilment for the award of degree of Master of Technology in Signal Processing & Digital Design at **Delhi Technological University, Delhi**, is a bonafide record of student's own work carried out by him under my supervision and guidance in the academic session 2014-15. To the best of my belief and knowledge the matter embodied in dissertation has not been submitted for the award of any other degree or certificate in this or any other university or institute.

Mr Rajesh Rohilla

Supervisor

Associate Professor

Dep. ECE

Delhi Technological University

Acknowledgement

I am indebted to my thesis supervisor **Mr. Rajesh Rohilla, Associate Professor** Department of Electronics and Communication, for his gracious encouragement and very valued constructive criticism that has driven me to carry out the project successfully.

I am greatly thankful to **Prof. Prem R. Chadda,** Head of Department (Electronics & Communication Engineering), entire faculty and staff of Electronics & Communication Engineering and friends for their continuous support, encouragement and inspiration in the execution of this "**Thesis"** work.

Finally I express my deep sense of gratitude to my parents who bestowed upon me their grace and were source of my inspiration and encouragement.

Gaurav Pawar

M.Tech (SPDD)

2K13/SPD/06

Dedicated to my Parents, Mentor, Friends

and

the Almighty…

Table of Contents

ABSTRACT

Occlusion handling is one of the most challenging problems in MOT (multiple object tracking). To deal with the problem of occlusion handling we proposed a novel approach based on neural network classifier with particle filter and HOG feature descriptors filter. Initial step in MOT is object detection which is done by background subtraction. After obtaining detected objects further neural network (NN) classifier is applied to verify that weather object detected is of our interest or not. Suppose if we want to track human, then with the help of NN classifier we can discard non-human objects. After object detection we will compute HOG feature descriptors for detected objects. But it is not robust to use all feature descriptors for feature matching as some of them are redundant. So to deal this problem we will assign weights to feature descriptors according to their ability of distinguishing one object from another. Feature descriptors which do not match with other object features or features which are usually similar to other object features should get assigned lower weights. And features which match only half of times to other object features should get assigned maximum weight. To track the objects we will use Particle filter as it can track objects which are moving in non-uniform pattern. Use of Particle filter will reduce region of interest for object matching.

To deal with the problem of occlusion handling we will do part based segmentation of object. If for example we use human as an object to be tracked, so in this case we will train a neural network classifier to segment each part of human body like head, hands and torso. So even if object is partially occluded, some part of the object still remains visible. That visible part is detected by Neural Network classifier and matching of that part is done with the corresponding part of other objects. If object is completely occluded or get disappeared then particle filter will be used to predict the location of that object.

Chapter 1 : Introduction

Object tracking is one of the most popular problems in Computer Vision. It has large number of applications like surveillance, tracking path of Missiles and Rocket, movement control of recording cameras and many more. Object tracking basically constitutes of estimation of trajectory of moving target object, which is done by updating the motion model of the target. For updating the motion model algorithms like Kalman filter, Particle filter are usually employed. Kalman filter is used in case of linear motion model, whereas particle filter deal with problem of non-linearity. But both require an observation model to finally estimate the value of target. Observation model basically consist of sensor output, but to decide which observation is useful or which is futile, we need to do comparison with target object. This is done by comparing the feature descriptors. As feature descriptor HOG feature descriptors has recently gain more popularity due its less computational requirement and its invariance against various changes like colour, scale and rotation. Observation model is based segmented objects that is really crucial for real time applications to avoid unwanted objects as target candidates. To identify relevant candidates recognition is done using classifier's that can recognize a desired candidate object in irrespective of its view. For this neural network classifiers are more popular as they can classify objects in more than 1 class. Now we will get brief review about classification, feature descriptors and tracking methods.

1.1. CLASSIFICATION

Classification is defined as 'ability of machine to classify a certain object into a category or class on the basis of features obtained from object'. And features are those characteristics of an object that differentiate it from other objects. For example, if we have to classify a pet animal as a cow or buffalo, so on the basis of its features like shape, size and color we can classify its either as a cow or buffalo. Classification is usually of two types: 1.Supervised 2.Unsupervised classification. In supervised learning first we trained a model, but in unsupervised learning model learn itself. In supervised learning we trained classifier according to the given features of object to be classified known as positive feature set and features of those objects which are different from object to be classified and which could appear in background of that object, these features are known as negative feature set. So by using positive and negative feature set classifier will able to model a decision boundary that can classify an object in two classes. Where as in unsupervised learning classifier uses an algorithm based on clustering, in which it makes cluster of similar kind of features. Neural networks, logistic regression and support vector machine are supervised classification techniques, k-mean clustering and mean-shift clustering are unsupervised classification methods.

As we have used neural networks for classification, so before discussing that we will discuss about classification using linear regression and maximum likelihood estimation. Then we will introduce ridge regression and regularization to fit non-linear functions by using bases functions.

1.1.1. LINEAR REGRESSION

Regression is defined as fitting of multiple points on a single line. So, in linear regression we are going to formulate a linear model on the basis of its given data points which can later predict the output (class) for those input data points. Linear regression is a supervised classification technique.

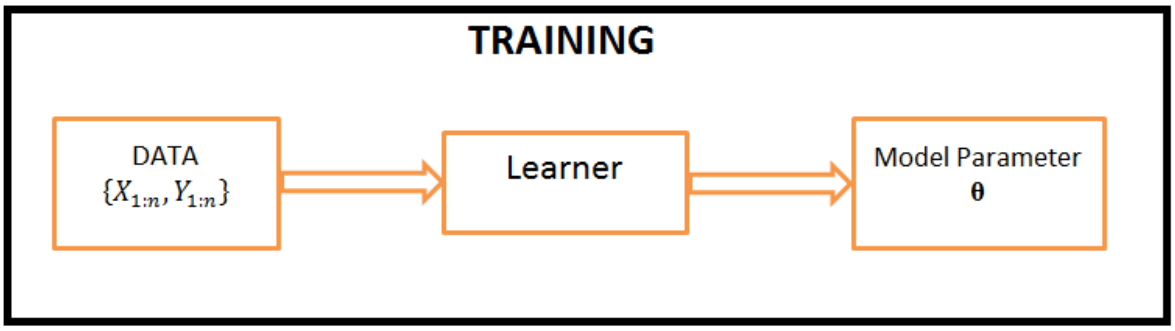

Figure 1.1: Training

1.1.1.1. Linear model

Formulation of linear model is a two-step process, training and testing. In training we have given a set of data points $\{X_{1:n}, Y_{1:n}\}$, where $X_{1:n}$ is a input set of data, like feature set of object of some particular dimension and $Y_{1:n}$ are output data points like output class for those input feature points. So, here $X_{1:n}$ is independent data variable and $Y_{1:n}$ is a scalar dependent data variable.

Training is a learning phase for a linear model, where with the help of given data points it learns the model parameters $\boldsymbol{\theta}$ which defines the relation between input data (feature vectors) points and output data (class of object) points.

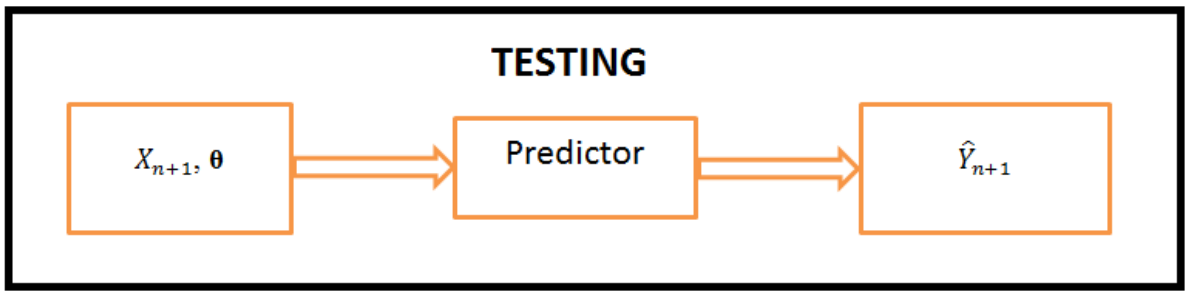

Figure 1.2: Testing

After obtaining model parameters we can predict output data points for input data points in testing phase. A linear model can be defined as follows:

$$\hat{Y}(X_i) = \theta_1 + X_i\theta_2$$

Where θ_1(intercept) and θ_2(slope) are linear model parameters. The difference between original output and predicted output is known as error. Our aim is to evaluate model parameters such that error is minimized. To achieve that optimization technique like least square error minimization can be used. By using least square error minimization minimum value of cost function will be evaluated. Here cost function is defined as follows:

$$J(\theta) = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 = \sum_{i=1}^{n}(Y_i - \theta_1 - X_i\theta_2)^2$$

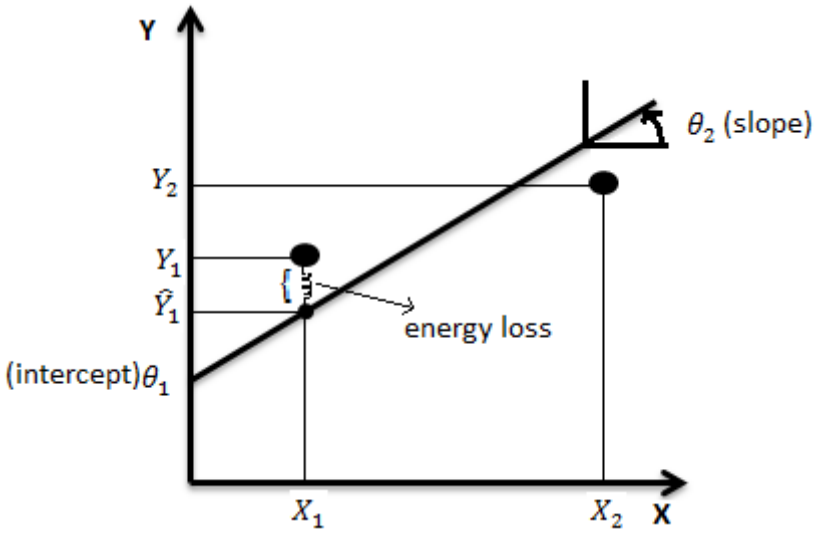

Figure 1.3: energy loss in linear model

Cost function also known as objective function or energy loss. Our objective is to minimize energy loss to obtain optimum values of model parameters. Generally a linear model is defined as:

$$\hat{Y}_i = \sum_{j=1}^{d} X_{ij}\ Q_j$$

In matrix form it can be written as:

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1d} \\ X_{21} & \ddots & X_{2d} \\ \vdots & \cdots & \vdots \\ X_{n1} & \cdots & X_{nd} \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_d \end{bmatrix}$$

$$J(\theta) = (\boldsymbol{Y} - \boldsymbol{X\theta})^T(\boldsymbol{Y} - \boldsymbol{X\theta}) \; = \sum_{i=1}^{n} (Y_i - \; {X_i}^T\theta)^2$$

Now minimize $J(\theta)$ by differentiating it w.r.t θ_1, θ_2.

$$\frac{\partial\, J(\theta)}{\partial \boldsymbol{\theta}} = \; \frac{\partial(Y^TY - 2Y^TX\theta + \theta^TX^TX\theta)}{\partial \boldsymbol{\theta}} = 0 - 2X^TY + 2X^TX\theta$$

Take $\frac{\partial\, J(\theta)}{\partial \boldsymbol{\theta}} \cong \; 0$

And we obtained model parameters as:

$$\boldsymbol{\theta} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$$

1.2. HOG Feature Descriptor Algorithm

1.2.1. Gradient computation

To find the horizontal and vertical gradients, convolve the image with kernels [-1 0 1] and [-1 0 1]^T.

1.2.2. Orientation binning

After computing gradients find out the orientations for each pixel gradients and then do binning for these into 9 classes from 0° to 180° with an interval of 20°. Number of pixels in a particular class constitutes the weight for that particular bin.

1.2.3. Descriptor blocks

To deal with the problem of changes in illumination and contrast, the gradient strength normalized locally, which need grouping of small cells into larger block. And finally we get HOG descriptor as a vector component of normalized cell histograms for complete block region. As blocks are overlapping, so each cell contributes more than once. In [1], suggested two kind of block geometry: R-HOG and C-HOG. In R-HOG object image is divided into 3×3 cells block, and cell has size of 6×6 pixels, and each cell has further 9 channels for per cell histogram.

C-HOG block exists in further two variants: one with single central cell and other with angular divided central cell. C-HOG block usually consists of four parameters: number of radial and bins, radius of center bin, and expansion factor for radius of additional radial bins.

In last step block normalization need to be done. There are four different methods for block normalization: L2-norm, L2-hys, L1-norm and L1-sqrt.

After calculation of HOG feature descriptors, SVM classifier is used for training. It is a binary classifier based supervised learning, which determines an optimal hyperplane as a decision function. In this way it can be used for object recognition in object tracking.

1.3. Particle Filter

Particle filter works on Bayesian model in which probability of hypothesis at time 't' is depend on hypothesis at time 't-1' and Observation data up to time 't'.

Bayes Model: P($H_t / H_{t-1}, D_t$)

In particle filter Monte Carlo simulation of data is done.

Our aim is to find out hypothesis at time 't', given hypothesis at time 't-1', action data at time 't' and observation data at time 't'. Here hypothesis could be the position and velocity of target object.

$$P(H_t / H_{t-1}, A_t, D_t) = ?$$

We model above motion model with particle samples. We generate particles around expected mean location.

After action command, we will update position of particles on the basis of model noise. Then we look for observations at updated location of particles.

For all the updated particles we will calculate the likelihood for observed data. In this way we update all the particles.

Now, we will do resampling.in resampling we will low weighted particles. And we get new mean value.

We have some prior distribution, and then on basis of action command we project particles to new location, and then weight them on basis of observation data and do resampling leading to updated distribution. And new estimated value is obtained.

Chapter 2 : Literature review

Introduction

Histogram of Oriented Gradients (HOG) is a feature descriptor used in computer vision and Image Processing in Object tracking for the purpose of object detection and object recognition. In HOG we count the occurrence of gradient orientation in particular localized area of an image. This method is similar to that of edge orientation histograms, scale invariant feature transform descriptors, and shape contexts, but differs in that it is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization for improved accuracy.

Histogram of Oriented Gradient descriptors first described by NavneetDalal and Bill Triggs[1], researchers for the French National Institute for Research in Computer Science and Control(INRIA), in their June 2005CVPR paper. The main idea behind the Histogram of oriented gradients is that local object appearance and shape with in an image can be described by the distribution of intensity gradients or gradient orientations. To implement these feature descriptors first we divide image into small connected regions, called cells and for each cell we compute histogram of gradient directions for all the pixels with in the cells. Then for all the cells combination of these histograms form the features descriptor.

2.1. Brief overview of HOG based tracking methods

We are going to discuss various HOG based Object tracking and classification techniques. We will discuss the original HOG descriptor algorithm proposed by NavneetDalal and Bill Triggs [1], and then we will explore other variants of HOG feature descriptor used for object tracking like 'Online multiple instance gradient feature selection' proposed by Yuan Xie [2], 'Moving object classification using local shape and HOG features in wavelet-transformed space' proposed by Chung-Wei Liang [3] and 'Interactive multiple model particle filter' proposed by Jianfang Dou [4].

In [2] a new gradient-based Histogram of Oriented Gradient (HOG) feature selection mechanism is employed under Multiple Instance Learning (MIL) framework for building target appearance model.The problem statement mainly consists of how to design an efficient appearance model and how to update it online in robust manner.

A gradient based feature selection approach is proposed with multiple instance learning for robust object tracking. Here a discriminative object appearance model used which consist of HOG rectangle featuresand their corresponding feature bags from positive and negative samples. Here appearance model updates robustly by iteratively updating features using

gradient decent and MIL approaches by maximizing the likelihood of training HOG feature bags.

Object tracking system consists of three steps: image representation, object appearance model and motion model. For object tracking first in current frame set of image patches are crop out. Some of image patches are put into positive bags and rest are in negative bags in accordance with the portion of object covered by patch. Then features are combined into a discriminative classifier. HOG is used as a Feature descriptor here. After this Online Gra-MIL classifier is employed to estimate the likelihood of bags in new frame. Location with maximum likelihood will be updated as a new location of object to be tracked. Then again two classes of bags of image patches are cropped out as positive and negative bags. And determine HOG features for those bags. Hence object appearance model get updated by using the feature descriptors of HOG rectangle bags.

In [3] adaptive background is used with RGB colour model, where each moving object is segmented with its minimum enclosing rectangle (MER) window by using histogram based projection approach or tracking based approach. For MER windows of different size object, window scaling operation followed by an adaptive block-shifting operation is applied to obtain features of fixed dimension. A weight mask is applied to MER window such that features vectors can categorized according to their ability of distinguishing it from other objects. To extract classification features, two-level Haar wavelet transformed is applied to MER window, then local shape features and Histogram of oriented gradients (HOG) features are calculated from level-two and level-one sub-bands, respectively, of wavelet transformed space. Finally a hierarchical linear support vector machine classifier is applied to classify the candidate objects into different classes.

In [4] multiple model based visual object tracking technique is used. Generally in visual object tracking single object model is employed. But as the environment around object to be tracked keep on varying and usually a particular object description model is not robust against all surrounding changes, to deal these issues here multiple model based visual object tracking technique is employed. So by integrating multiple cues, interactive multiple models (IMM) are combined with particle filter to form IMM_PF filter resulting in to a dynamic system which can adapt to various tracking scenarios like partial occlusion, rotation and abrupt shift. IMM consists of three observation models: Histogram of Oriented Gradients (HOG), Corrected Background Weighted Histogram (CBWH) and Completed Local Ternary Patterns (CLTP). Likelihood of a particular observation model decides it contribution to the final tracking result.

2.2. Brief overview of Particle filter based tracking methods

In [7] authors propose an effective algorithm for multiple pedestrians tracking, which is constructed in the framework ofparticle filtering, and it is based on the combination of online boosting tracker and the histogram of oriented gradient (HOG)descriptor for human detection. The combination for the detector and tracker lies on following aspects. First, each detectionresult is associated to a tracker implemented by the online boosting, which gives the authors scheme robustness for multiplesimilar objects and then, the output of support vector machine classifier based on HOG is dynamically fused as a componentin the observation metric in particle filtering, which makes the tracker more accurate in some difficult conditions. Finally, thestates of some particles are replaced by the state given by the detector, so that the tracker can recover from failure quickly.

In [8] the detection and tracking of an unknown number of targets using a Bayesian hierarchical model with target labelsare presented. To approximate the posterior probability density function (PDF), we develop a two-layer particle filter (PF). One deals with track initiation, and the other deals with track maintenance. In addition the parallel partition (PP) method is proposed to sample the states of the surviving targets.

In [9] authors propose a vision-based automatic system to detect preceding vehicles on the highway under various lighting and different weather conditions. To adapt to different characteristics of vehicle appearance under various lighting conditions, fourcues including underneath shadow, vertical edge, symmetry and taillight are fused for the vehicle detection. The authors achievethis goal by generating probability distribution of vehicle under particle filter framework through the processes of initial sampling,propagation, observation, cue fusion and evaluation. Unlike normal particle filter focusing on single target distribution in a statespace, the authors detect multiple vehicles with a single particle filter through a high-level tracking strategy using clustering. Inaddition, the data-driven initial sampling technique helps the system detect new objects and prevent the multi-modal distribution from collapsing to the local maxima.

In [10] a particle filter (PF) has been proposed to detect and track colour objects in video. This study presents anadaptation of the PF to track people in surveillance video. Detection is based on automated background modelling rather thana manually generated object colour model. Furthermore, a labelling method is proposed to create tracks of objects through thescene, rather than unconnected detections. The PF tracker gives significantly fewer false alarmsowing to explicit modelling of the object birth and death processes, while maintaining a good detection rate.

In [11] author develop a novel solution for particle filtering on general graphs. We provide an exact solution for particle filtering on directed cycle-free graphs. The proposed approach relies on a partial-order relation in an antichain decomposition that forms a high-order Markov chain over the partitioned graph. We subsequently derive a closed-form sequential updating scheme

for conditional density propagation using particle filtering on directed cycle-free graphs. They also provide an approximate solution for particle filtering on general graphs by splitting graphs with cycles into multiple directed cycle-free subgraphs. They then use the sequential updating scheme by alternating among the directed cycle-free subgraphs to obtain an estimate of the density propagation.They rely on the proposed method for particle filtering on general graphs for two video tracking applications: 1) objecttracking using high-orderMarkov chains; and 2) distributed multiple object tracking based on multi-object graphical interaction models.

In [12] author formulate object tracking in a particle filter framework as a structured multi-task sparse learning problem, which is denote as Structured Multi- Task Tracking (S-MTT). Since we model particles as linear combinations of dictionary templates that are updated dynamically, learning the representation of each particle is considered a single task in Multi-Task Tracking (MTT). By employing popular sparsity-inducing mixed norms and regularize the representation problem to enforce joint sparsity and learn the particle representations together. As compared to previous methods that handle particles independently, results demonstrate that mining the interdependencies between particles improves tracking performance and overall computational complexity. They extend the MTT framework to take into account pairwise structural correlations between particles (e.g. spatial smoothness of representation) and denote the novel framework as S-MTT. The problem of learning the regularized sparse representation in MTT and S-MTT can be solved efficiently using an Accelerated Proximal Gradient (APG) method that yields a sequence of closed form updates. As such, S-MTT and MTT are computationally attractive. They test proposed approach on challenging sequences involving heavy occlusion, drastic illumination changes, and large pose variations.

2.3. Brief overview of other popular tracking methods

Robust detection and tracking of multiple people in cluttered and crowded scenes with severe occlusion is a significant challenge for many computer vision applications. In [13] present a novel hybrid synthetic aperture imaging model to solve this problem. The main characteristics of this approach are as follows. 1) To the best of our knowledge, this is the first attempt to solve the occluded people imaging and tracking problem in a joint multiple camera synthetic aperture imaging domain. 2) A multiple model framework is designed to achieve seamless interaction among the detection, imaging and tracking modules. 3) In the object detection module, a multiple constraints-based approach is presented for people localization and ghost objects removal in a 3-D foreground silhouette synthetic aperture imaging volume. 4) In the synthetic imaging module, a novel occluder removal-based synthetic imaging approach is proposed to significantly improve the imaging quality of objects even under severe occlusion. 5) In the object tracking module, a camera array is used for robust people tracking in color synthetic aperture images. A network-camerabased hybrid synthetic aperture imaging system has been set up, and experimental results with qualitative and quantitative analyses demonstrate that the method can reliably locate and see people in challenging scenes.

Detecting multiple targets and obtaining a record of trajectories of identical targets that interact mutually infer countless applications in a large number of fields. However it presents a significant challenge to the technology of object tracking. In [14] describes a novel structured learningbased graph matching approach to track a variable number of interacting objects in complicated environments. Different from previous approaches, the proposed method takes full advantage of neighboring relationships as the edge feature in a structured graph, which performs better than using the node feature only. Therefore, a structured graph matching model is established, and the problem is regarded as structured node and edge matching between graphs generated from successive frames. In essence, it is formulated as the maximum weighted bipartite matching problem to be solved using the dynamic Hungarian algorithm, which is applicable to optimally solving the assignment problem in situations with changing edge costs or weights. In the proposed graph matching model, the parameters of the structured graph matching model are determined in a stochastic learning process. In order to improve the tracking performance, bilateral tracking is also used. Finally, extensive experimental results on Dynamic Cell, Football, and Car sequences demonstrate that the new approach effectively deals with complicated target interactions.

In [15], author show that integrating such prior information into a supervised learning algorithm can handle visual drift more effectively and efficiently than the existing MIL tracker. They present an online discriminative feature selection algorithm that optimizes the objective function in the steepest ascent direction with respect to the positive samples while in the steepest descent direction with respect to the negative ones. Therefore, the trained classifier directly couples its score with the importance of samples, leading to a more robust and efficient tracker.Most tracking-by-detection algorithms train discriminative classifiers to separate target objects from their surrounding background. In this setting, noisy samples are likely to be

included when they are not properly sampled, thereby causing visual drift. The multiple instance learning (MIL) paradigm has been recently applied to alleviate this problem. However, important prior information of instance labels and the most correct positive instance (i.e., the tracking result in the current frame) can be exploited using a novel formulation much simpler than an MIL approach.

Motivated by the active learning method, in [16]author propose an active feature selection approach that is able to select more informative features than the MIL tracker by using the Fisher information criterion to measure the uncertainty of the classification model. Adaptive tracking by detection has been widely studied with promising results. The key idea of such trackers is how to train an online discriminative classifier, which can well separate an object from its local background. The classifier is incrementally updated using positive and negative samples extracted from the current frame around the detected object location. However, if the detection is less accurate, the samples are likely to be less accurately extracted, thereby leading to visual drift. Recently, the multiple instance learning (MIL) based tracker has been proposed to solve these problems to some degree. It puts samples into the positive and negative bags, and then selects some features with an online boosting method via maximizing the bag likelihood function. Finally, the selected features are combined for classification. However, in MIL tracker the features are selected by a likelihood function, which can be less informative to tell the target from complex background. More specifically, propose an online boosting feature selection approach via optimizing the Fisher information criterion, which can yield more robust and efficient real-time object tracking performance.

In [17]author presents an innovative method, which uses projected gradient to facilitate multiple kernels, in finding the best match duringtracking under predefined constraints. The adaptive weights are applied to the kernels in order to efficiently compensate theadverse effect introduced by occlusion. An effective scheme is also incorporated to deal with the scale change issue during the object tracking. Moreover, we embed the multiple-kernel tracking into a Kalman filtering-based tracking system to enable fully automatic tracking. Several simulation results have been done to show the robustness of the proposed multiple-kernel tracking and also demonstrate that the overall system can successfully track the video objects under occlusion.Kernel based trackers have been proven to be a promising approach for video object tracking. The use of a single kernel often suffers from occlusion since the available visual information is not sufficient for kernel usage. In order to provide more robust tracking performance, multiple inter-related kernels have thus been utilized for tracking in complicated scenarios.

In [18] author presents an extended Markov chain Monte Carlo (MCMC) method for tracking and an extended hidden Markov model (HMM) method for learning/recognizing multiple moving objects in videos with jittering backgrounds. A graphical user interface (GUI) with enhanced usability is also proposed. Previous MCMC and HMM-based methods are known to suffer performance impairments, degraded tracking and recognition accuracy, and higher computation costs when challenged with appearance and trajectory changes such as occlusion, interaction, and varying numbers of moving objects. This paper proposes a cost reduction method for the MCMC approach by taking moves, i.e., birth and death, out of the iteration loop of the Markov chain when different moving objects interact. For stable and robust tracking, an ellipse model with stochastic model parameters is used. Moreover, our HMM method integrates several different modules in order to cope with multiple discontinuous trajectories. The GUI proposed herein offers an auto-allocation module of symbols from images and a hand-drawing module for efficient trajectory learning andfor interest trajectory addition.

In [19] a fully automatic multiple-object tracker based on mean-shift algorithm is presented.Mean-shift tracking plays an important role in computer vision applications because of its robustness, ease of implementation and computational efficiency. Foreground is extracted using a mixture of Gaussian followed by shadow and noise removal to initialise the object trackers and also used as a kernel mask to make the system more efficient by decreasing the search area and the number of iterations to converge for the new location of the object. By using foreground detection, new objects entering to the field of view and objects that are leaving the scene could be detected. Trackers are automatically refreshed to solve the potential problems that may occur because of the changes in objects' size, shape, to handle occlusion-split between the tracked objects and to detect newly emerging objects as well as objects that leave the scene. Using a shadow removal method increases the tracking accuracy. As a result, a method that remedies problems of mean-shift tracking and presents an easy to implement, robust and efficient tracking method that can be used for automated static camera video surveillance applications is proposed.

In [20] author proposes an object tracking algorithm that learns a set of appearance models for adaptive discriminative object representation. In this paper, object tracking is posed as a binary classification problem in which the correlation of object appearance and class labels from foreground and background is modelled by partial least squares (PLS) analysis, for generating a low-dimensional discriminative feature subspace. As object appearance is temporally correlated and likely to repeat over time, we learn and adapt multiple appearance models with PLS analysis for robust tracking. The proposed algorithm exploits both the ground truth appearance information of the target labelled in the first frame and the image observations obtained online, thereby alleviating the tracking drift problem caused by model update.

Object appearance modelling is crucial for tracking objects, especially in videos captured by no stationary cameras and for reasoning about occlusions between multiple moving objects. Based on the log-euclidean Riemannian metric on symmetric positive definite matrices, In [21] author propose an incremental log-euclidean Riemannian subspace learning algorithm in which covariance matrices of image features are mapped into a vector space with the log-euclidean Riemannian metric. Based on the subspace learning algorithm, we develop a log-euclidean block-division appearance model which captures both the global and local spatial layout information about object appearances. Single object tracking and multi-object tracking with occlusion reasoning are then achieved by particle filteringbased Bayesian state inference. During tracking, incremental updating of the log-euclidean block-division appearance model captures changes in object appearance. For multi-object tracking, the appearance models of the objects can be updated even in the presence of occlusions.

In [21]author considers the problem of jointly detecting whether a target is present in a scene and estimating its state, if it is there. This joint detection and estimation problem can be solved using a special case of the multi-target Bayes filter (referred to as the joint target detection and tracking (JoTT) filter). However, if the model used by the JoTT filter does not match the actual dynamics, the filter will tend to miss-detection directly or diverge such that the actual errors fall outside the range predicted by the filter's estimate of the error covariance. A similar difficulty arises, if the target behaviour can switch between different modes of operation, since the filter may then be accurate for only one particular mode. This study proposes a novel joint detection and tracking filter, which is the multiple model extension of the JoTT filter to accommodate the possible target manoeuvring behaviour. In addition, a sequential Monte Carlo implementation (for generic models) and a Gaussian mixture implementation (for linear Gaussian models) are proposed.

Chapter 3 : Proposed Method

To handle the problem of Occlusion handling in multiple object tracking we introduced the concept of Part base segmentation. That is when an object is occluded with other object then in object segmentation step it is detected as a single object, so to identify whether object detected is occluded or single object we look for the count of some part of object to be detected. For example face count in case of human detection, so in case of partial occlusion we will obtain more than 1 face in segmented object.For Feature descriptors we have used HOG descriptors followed by particle filter for tracking. For part based segmentation we have employed neural network based classifier.

Algorithm proposed for tracking consists of following steps:

Figure 2.1: sample of front view database

Figure 3.2: sample images of back view database

3.1. Database formation

As we are using neural network classifier for object segmentation, so before recognize any image you need to do training for a classifier. For example, if object to be tracked is a human then we need to train our classifier for different views of human. So we need database fordifferent views of object to be tracked. Usually a database consists of positive and negative set of images. Positive set consist of images of object to be tracked and negative set consist of images of all other objects that could appear in background of the object to be tracked.

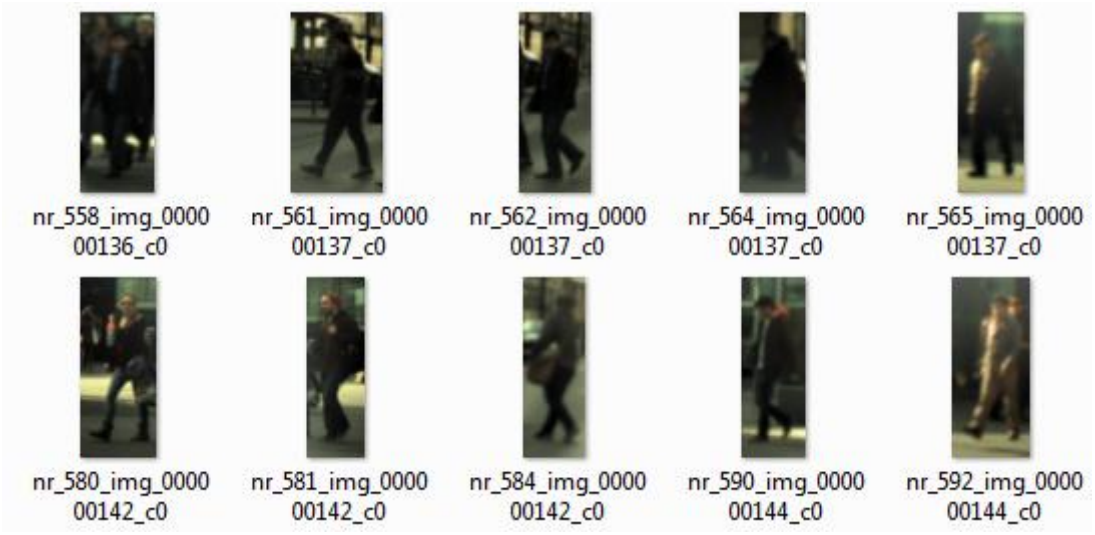

Figure 3.3: sample images for left view database

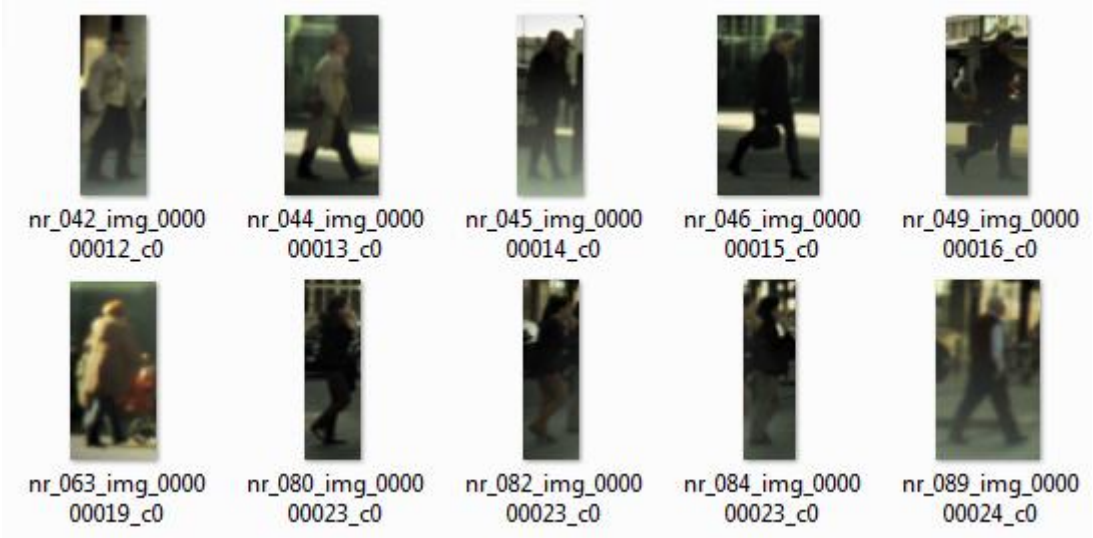

Figure 3.4: sample images of right view database.

3.2. Classifier training

For training a classifier we need feature descriptor set for both positive and negative set of images. Ideally we need such a feature descriptors which should converge well for a single class and should be invariant of surrounding changes. So we are using HOG feature descriptors for training negative and positive imagesof dataset. Here we have used neural network based classifier for classifier training. In neural network classifier we have used two variants: 1. Radial basis function neural network (RBFNN) based classifier, 2. Probabilistic neural network (PNN) classifier. But in implementation in main algorithm we preferred Probabilistic neural network classifier because it gives exact 0 and 1 as output class, but in RBFNN classifier we get values between 0 to 1, so it is difficult to decide threshold value to classify it in a particular class.

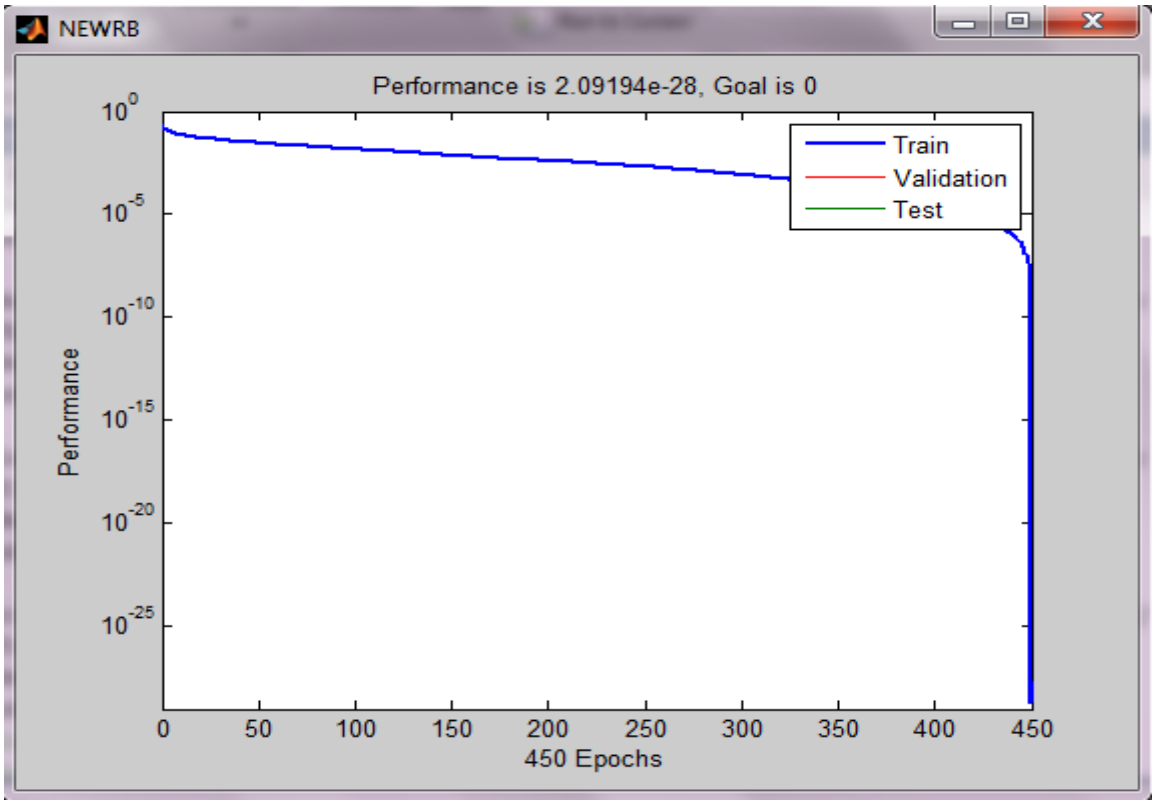

Figure 3.5: Result for training RBFNN classifier for Front view class images.

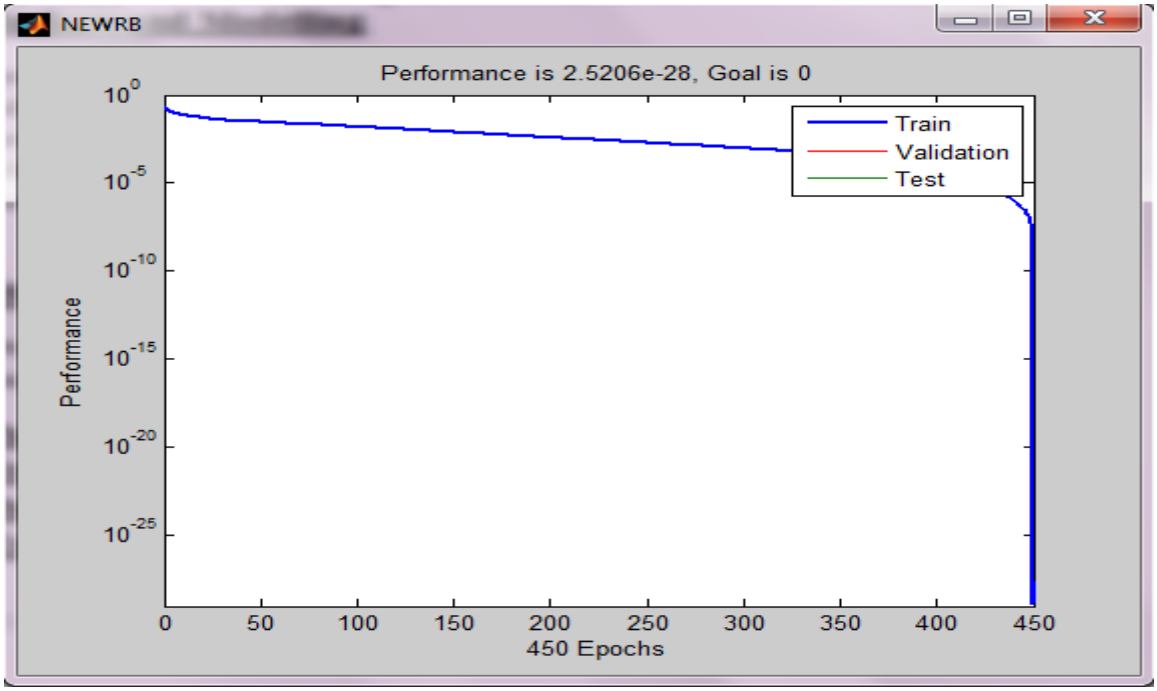

Figure 3.6: Result for training RBFNN classifier for Back view class images.

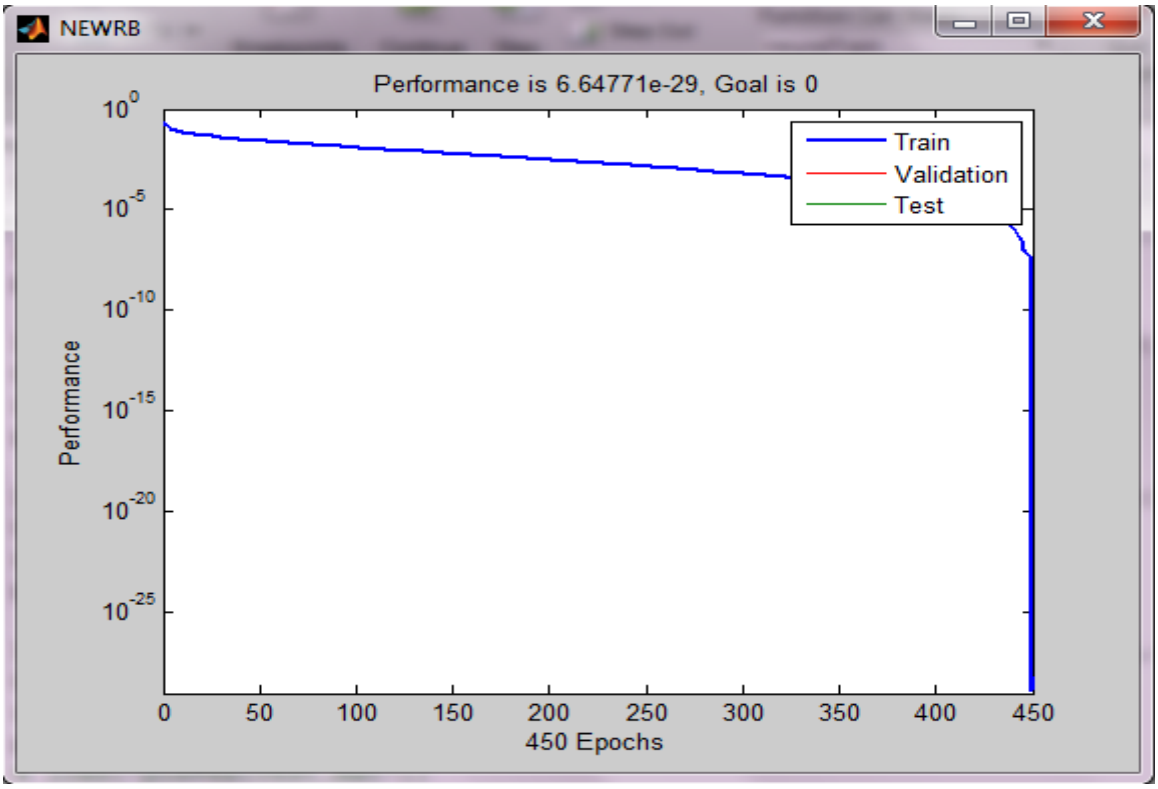

Figure 3.7: Result for training RBFNN classifier for Left view class images.

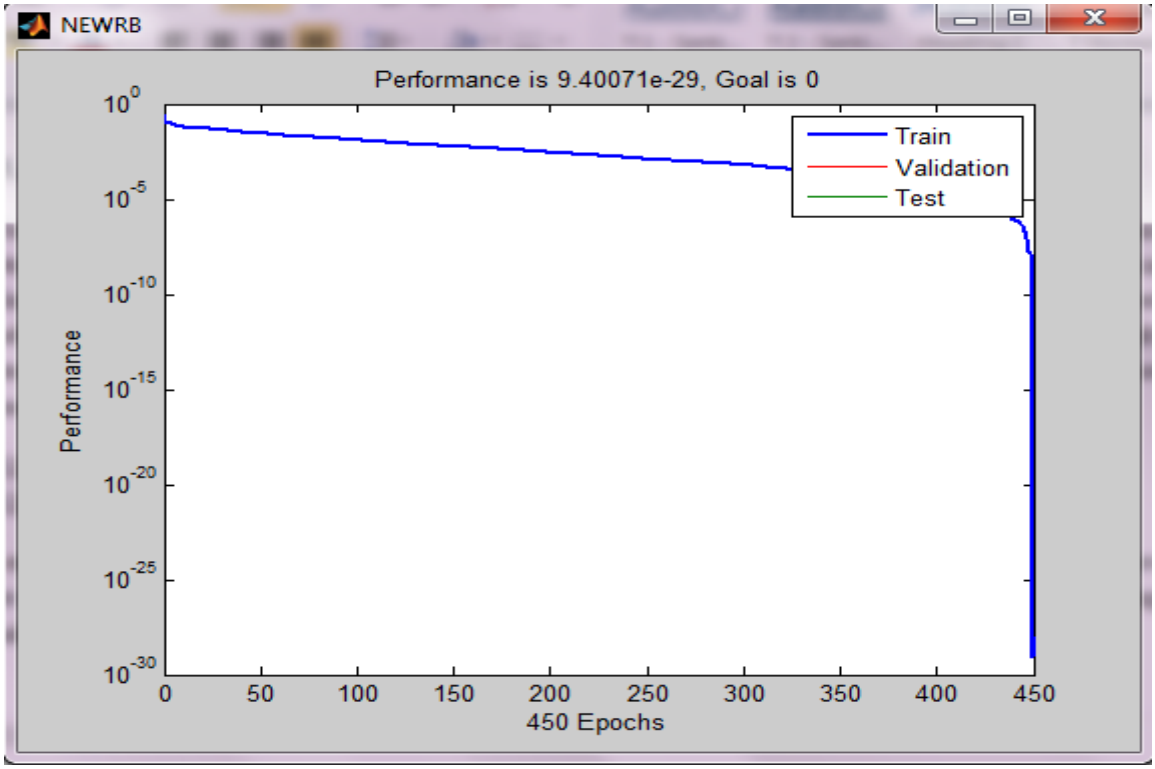

Figure 3.8: Result for training RBFNN classifier for Right view class images.

3.3. Background Modelling

In background modelling we determine the background in which object need to be tracked. To obtain background we are using averaging of frames such that it can discard the changes occurs due to moving object, and as time passes it will get updated because it keep on including new frames with time. In this way we obtained adaptive background model that adapt itself with environment change and lead to robust results.

Figure 3.9: Result of background modeling and background subtraction.

Figure 3.10: Result of background modeling and background subtraction.

3.4. Object Segmentation

After obtaining a background, to obtain objects to be tracked we will do background subtraction. After doing subtraction we decide a threshold value and the pixels for which we get difference more than threshold value will get assigned value binary 1 and rest will get assigned value binary zero. In this way get a binary image output for background subtraction. Object Segmentation main aim is to obtain localization and geometrical information of segmented objects. But after background subtraction we got a sort of connected pixels with clutter associated it with, so to remove clutter and obtain objects properly we apply some morphological operations like erosion, dilation and holes filling. First we will discuss these morphological operations in brief as follows:

3.4.1. Morphological operators

3.4.1.1. Erosion

In erosion we eliminate a pixel or make it value binary 0, if all of its neighbour pixels have zero magnitude. The neighbour pixels selection is depend on the kind of mask you are using. Mask could be of following shape: 1. Square, 2. Oval, 3. Circle, 4. Line and many more. The aim of performing erosion operation is to supress the clutter.

3.4.1.2. Dilation

In dilation we perform the opposite operation as compare in erosion. In dilation aim is to recover the lost information due to background and object similarity and due to noise. In dilation a pixel is converted to binary 1, if any of its neighbours has binary value 1. And its neighbour pixels are decided by the geometry of mask used as in erosion.

3.4.1.3. Hole filling

Even after performing erosion and dilation some hole are left in object, so to fill those holes we employed this process. It is similar to dilation, but a recursive process.

After performing morphological operations, still localized and geometrical information of connected set of pixels is unknown. To obtain that information we perform another operation known as blob analysis. In blob analysis if iterate the complete image and look for connected

set of pixel which has magnitude binary 1. And it will then parameters like area, corner point location, centroid and width and height of those set of connected high magnitude pixels.

3.4.2. Classifier based recognition

After doing background subtraction followed by some morphological operations and blob analysis we obtain objects, but it is not assured that those objects are valid candidate objects for tracking. So to find out whether the object detected is of our interest or not we use a trained classifier. That classifier is trained to recognize the object of our interest. So here we are using PNN classifier. We trained classifier for each possible view of candidate object, because a moving object will change its view with motion. But a single classifier cannot be trained for all possible views of an object. So we have to train our classifier for all the different views. That means we have to prepare database for each possible view as a different class database. And then train classifier for multiple classes. In this way we can discard objects which are not of our interest.

3.4.3. Part base segmentation

After obtaining connected pixels as an object in object segmentation, it might happen some times that more than one object appear to be a single object. It usually occurs when objects get occluded or objects are too close to each other. So to deal with this problem of separating objects from an occluded image we need to use part base segmentation. In this method we instead of looking for complete object we will look for parts of that object. So for doing part base segmentation we have to train a classifier for those parts of object. And then we iterate the object image through a sliding window of variable size, and then for each object extracted corresponding to sliding window we apply PNN classifier to recognize that whether that object is that desired part of object or not. For example if we trained our classifier for face of an human, so in case if there are more than one faces in that object then that indicate same number of that objects in that image and we have to again apply same procedure to extract that object as we have done for extracting part of that object. That is we have to do sliding with variable size of window and then do recognition that whether that object is human or not.

3.5. Feature descriptor calculation

Feature descriptors we have used to characterize an object are HOG [5] feature descriptors and its variant RDHOG [5] feature descriptors. These feature descriptors are well efficient and robust and also less computationally expensive in compare to features descriptors like SURF and SIFT. And these are also invariant to changes like scale, rotation and colour. HOG abbreviated as histogram of oriented gradients and RDHOG as relative discriminative HOG.

3.5.1. HOG Feature descriptors

To calculate HOG feature descriptors first we resize our object to 32 by 32 pixels size. Then divide that in 9 overlapping blocks of 16 by 16 pixels size. Then each block is further divided into 4 cells of 8 by 8 pixels size. Then for each cell gradient is calculated, gradient id defined as change in intensity levels in particular direction. Then orientation for those gradients will be

calculated and that orientation is further classified in to nine classes. That means we are constructing histogram of nine bins with a step size of 20 degree as range of orientation value is 0 to 180 degree. So for each cell we are getting a vector of length 9, which means we will get a feature vector for 32 by 32 pixels image of length 324.

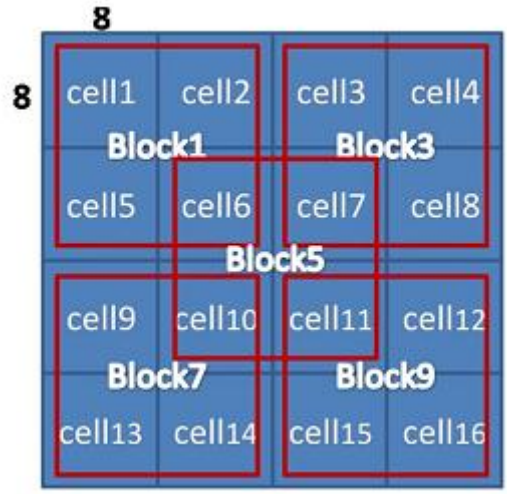

Figure 3.11: Block and Cell division in HOG and RDHOG feature descriptors

3.5.2. RDHOG Feature descriptors

After getting 324 length HOG feature descriptor, now we will compute RDHOG feature descriptor by comparing HOG features of each block with the central block, as there are 9 blocks so 8 comparisons will be done. So we will get RDHOG feature vector of length 288.

3.6. Data association

In data association we will decide which objects to take as candidate objects. And then find the target object from the candidate objects by doing feature matching of target and candidate objects. Suppose at time t we have only 2 objects in frame, but at time t + 1, 2 more objects come into picture. Then if we have to determine which object is object '2' in frame t+1, then first we look for candidate objects lie in proximity of last location of object '2', suppose 2 objects lie in that range, those two objects are candidate objects. Now to select the object as a target we compare the features of target object and candidate objects. And the candidate object for which we get maximum similarity measure will be considered as the target object. For similarity measure we used features descriptor like HOG and RDHOG. HOG is used to determine similarity and RDHOG for dissimilarity.

3.7. Tracking using Particle filter

After doing segmentation of object, we need to determine its motion model such that if it get disappears, so we can predict its location based on motion model. In motion mode we used parameters position and velocity. Now after getting motion model or state model, we need to update that model, so to do that update we use filters like Kalman and Particle filters [6].

Kalman filter is used for linear mode. That is the object moves with uniform and linear motion are tracked by Kalman filter. But in case of non-linear and non-uniform model we need to use particle filter. It is bit more computationally expensive but more efficient in compare to Kalman filter especially in non-linear motion model. The first step of particle filter after modelling motion mode is generation of particles.

3.7.1. Generate Particles

As object is moving in random direction we can predict its location, so to deal with that we use multiple particles to find the optimum location. The particles will be randomly generated in the range in which it is possible to find the target.

3.7.2. Update particles

After initializing particles we initially provide equal weightage to all the particles. Then we update the location of particles according to the motion model. After updating the location of all particles we will find the likelihood of all the particles with help of measurement data, that is we determine which particle resembles more similar to target object and accordingly assign weight to these particles and then we will take weighted mean to obtain the final estimated location of target object.

3.7.3. Particle Resampling

After updating particles with motion model assigning weight to them, now we will retain only those particles which have higher weights and discard rest of particles. So this process of retaining useful particles is called particle resampling. And after obtaining resampled particles we update our model with respect to the position and velocity obtained by those resampled particles.

But after certain iteration we get left with few numbers of particles this problem is called particle degeneracy problem which can be overcome by using genetic algorithms like Particle Swarm optimisation by optimising the location of particles during their sampling.

Chapter 4 :Results

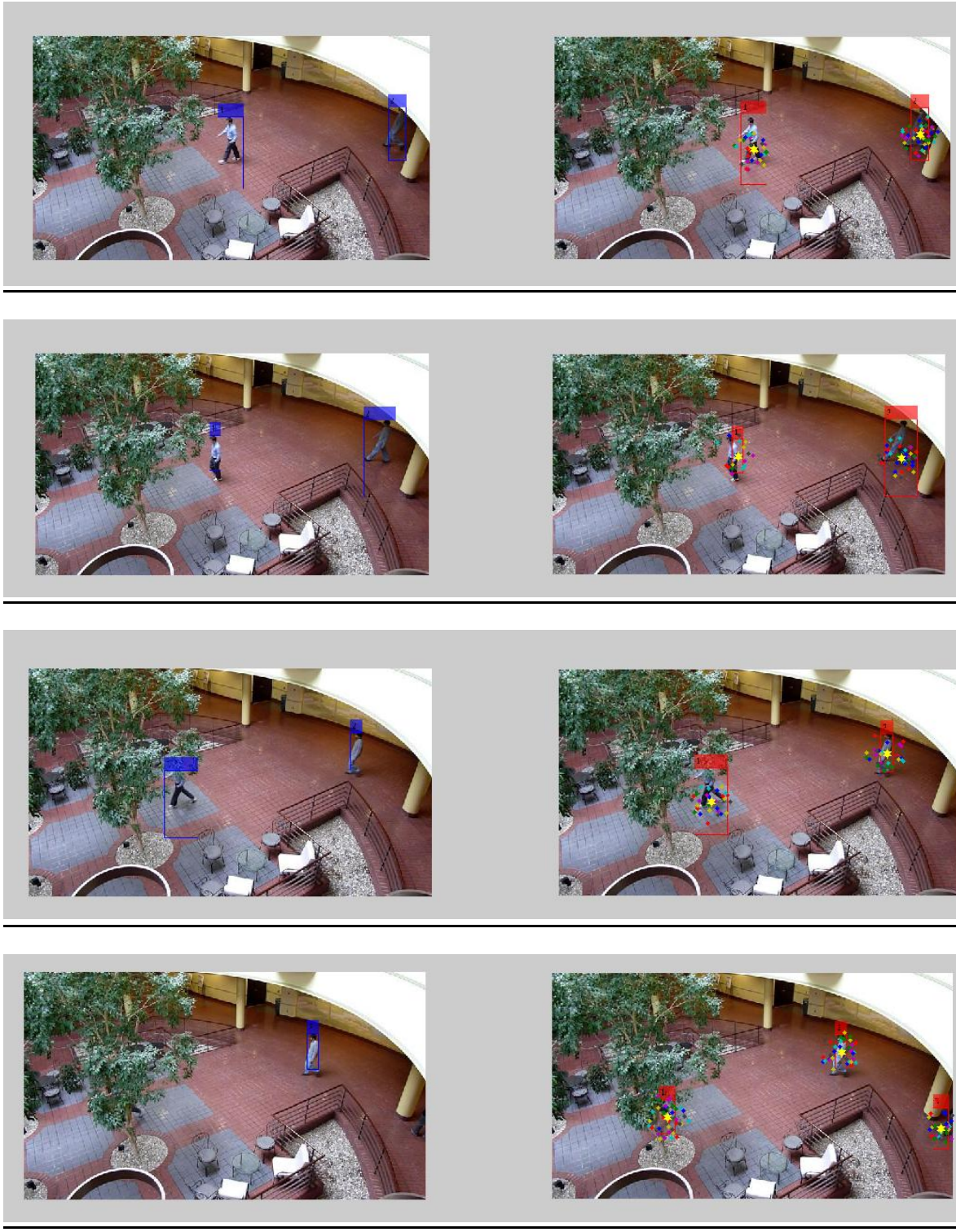

Figure 4.1: Tracking results where left image shows sensor output and right side image is estimated output.

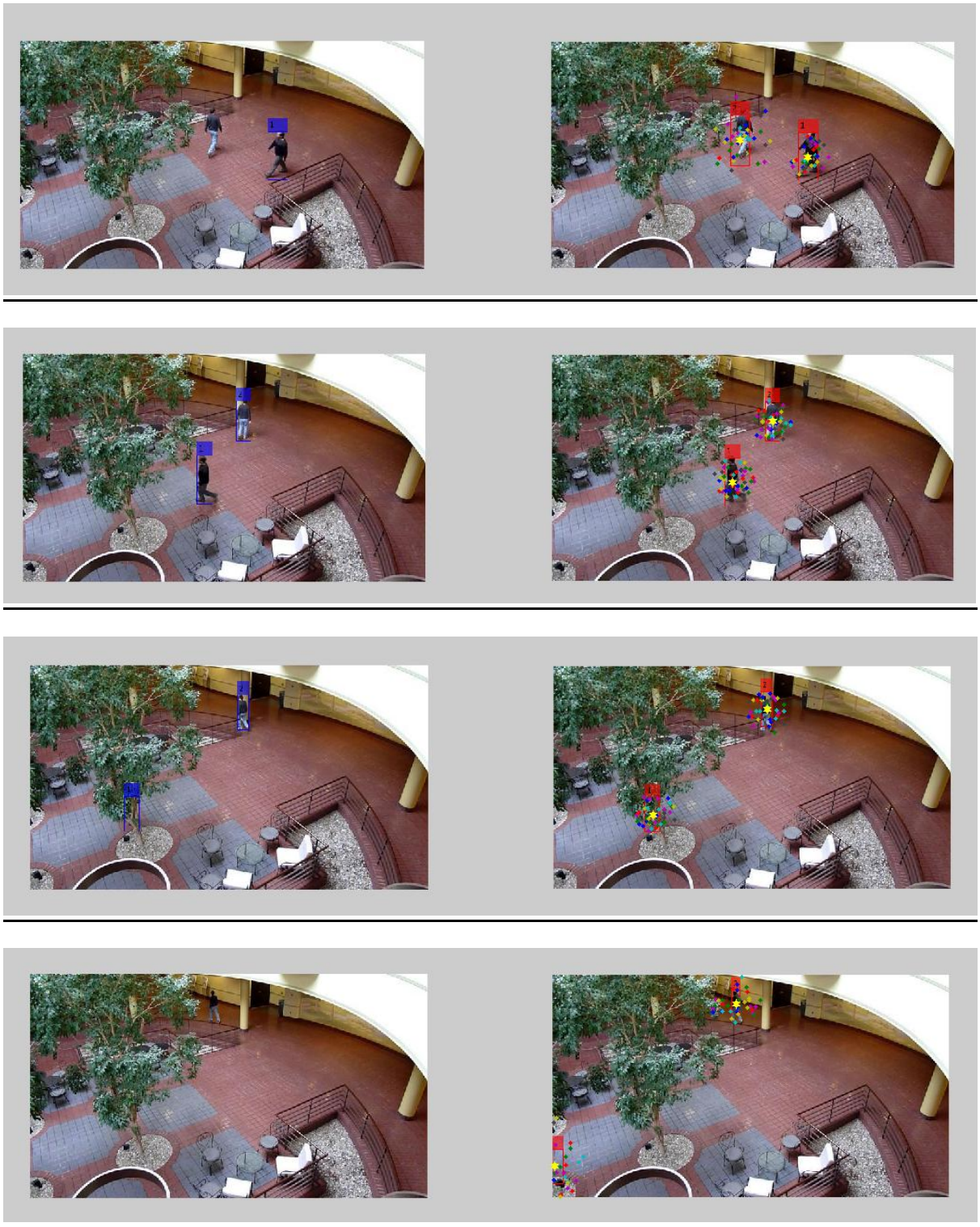

Figure 4.2:Tracking results where left image shows sensor output and right side image is estimated output.

Chapter 5 : Conclusion

- In this work we have implemented tracking algorithm based on particle filter which is tracking multiple objects, as a result of which it can handle motion of target object in random directions very efficiently.
- For feature descriptor we have used HOG feature descriptors and its variant RDHOG. While training of a classifier we have used HOG descriptor only to reduce the length of feature descriptor such that number of input node in input layer of neural network classifier does not get large number of input nodes.But in data association while feature matching we have used both HOG and RDHOG feature descriptors such it recognize the target correctly.
- In part based segmentation to handle occlusion we have used neural network classifier for object detection because it can classify more than 1 class as object detected could exist in different views.
- This method get fails in case of moving camera, or when object become stationary for a long time, and when objects move closely in a group.
- In future we can work on concurrent neural network classifiers and deep learning to improve performance of object detection as detection is crucial for efficient tracking.

References

[1]. Dalal, N., Triggs, B., 2005.Histograms of oriented gradient for human detection.In:Proc. Computer Vision and Pattern Recognition, pp. 886–893.

[2]. Yuan Xie, YanyunQu ,Cuihua Li , Wensheng Zhang. Online multiple instance gradient feature selection for robust visual tracking. Proc.Of the Pattern Recognition Letters 33 (2012) 1075–1082

[3]. Chung-Wei Liang, Chia-FengJuang. Moving object classification using local shape and HOG features inwavelet-transformed space with hierarchical SVM classifiers. Proc. oftheApplied Soft Computing 28 (2015) 483–497

[4]. Jianfang Dou, Jianxun Li. Robust visual tracking based on interactive multiple model particlefilter by integrating multiple cues. Proc. oftheNeurocomputing 135 (2014) 118–129

[5]. Bing-Fei Wu, Chih-Chung Kao, Cheng-Lung Jen,Yen-Feng Li, Ying-Han Chen, and Jhy-Hong Juang.A Relative-Discriminative-Histogram-of-OrientedGradients-Based Particle FilterApproach toVehicle Occlusion Handling and Tracking.Proc.Of the ieeetransactionson industrial electronics, vol. 61, no. 8, august 2014

[6]. N.J Gordan, D.J salmond,AFM smith. Novel approach to non-linear Bayesian state estimationProc. Ofthe IEE ,vol 140 no.2. 1993

[7Li Sun, Guizhong Liu, Yiqing Liu. Multiple pedestrians tracking algorithm byincorporating histogram of oriented gradientdetections.IET Image Process., 2013, Vol. 7, Iss. 7, pp. 653 –659

[8]. AN GEL F. GA RCIA-FERN´ ANDEZ´JESUS GRAJAL, MARK R. MORELANDE.Two- Layer Particle Filter for Multiple Target Detection andTracking. Proc. oftheieee transactions on aerospace and electronic systems vol. 49, no. 3 july 2013

[9]. Y.-M. Chan1,S.-S. Huang,L.-C.Fu,P.-Y.Hsiao,M.-F. Lo. Vehicle detection and tracking under various lightingconditions using a particle filter.Proc. oftheIET Intell. Transp. Syst., 2012, Vol. 6, Iss. 1, pp. 1–8

[10]. J. SherrahB. RisticN.J. Redding. Particle filter to track multiple people forvisual surveillanceProc. oftheIET Comput. Vis., 2011, Vol. 5, Iss. 4, pp. 192–200

[11]. Pan Pan and Dan Schonfeld. Video Tracking Based on Sequential ParticleFiltering on Graphs.Proc. oftheieeetransactionsonimage processing ,vol . 2 0 , no . 6 ,june 2 0 11

[12]. TianzhuZhang, Bernard Ghanem, Si Liu and NarendraAhuja. Robust Visual Tracking via Structured Multi-Task SparseLearning.International Journal of Computer Vision.

[13]. Tao Yang, Yanning Zhang, Xiaomin Tong, Xiaoqiang Zhang, and Rui Yu. A NewHybrid Synthetic Aperture Imaging Modelfor Tracking and Seeing People Through Occlusion.Proc. Ofthe ieee transactions on circuits and systems for video technology, vol. 23, no. 9, september2013.

[14]. HongkaiXiong, DayuZheng, Qingxiang Zhu, Botao Wang, and Yuan F. Zheng. AStructured Learning-Based Graph MatchingMethod for Tracking Dynamic Multiple Objects.ieee transactions on circuits and systems for video technology, vol. 23, no. 3, march 2013