A Project Report
On

REAL TIME ROAD TRAFFIC PREDICTION

Submitted in partial fulfilment of the requirements for the award of the degree of

MASTER OF TECHNOLOGY

In

VLSI AND EMBEDDED SYSTEM

SUBMITTED BY
SHRADDHA CHAUDHARY

Roll No: 13/VLSI/2010

UNDER THE GUIDANCE OF
Dr. S.INDU
(ASSOCIATE PROFESSOR)

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(FORMERLY DELHI COLLEGE OF ENGINEERING)
BAWANA ROAD DELHI-110042

2010-2012

CERTIFICATE

This is to certify that **Ms. Shraddha Chaudhary (13/VLSI/10)** has carried out the major project titled **"Real Time Road Traffic Prediction"** as a partial requirement for the award of Master of Technology degree in VLSI and Embedded Systems by Delhi Technological University.

The major project is a bonafide piece of work carried out and completed under my supervision and guidance during the academic session **2010-2012**. The matter contained in this report has not been submitted elsewhere for the award of any other degree.

(Project Guide)
Dr S.Indu
Associate Professor
Department of Electronics and Communication
Delhi Technological University
Bawana Road, Delhi-110042

ACKNOWLEDGEMENT

I express my gratitude to my major project guide Dr. S.Indu , Associate Professor ECE Dept., Delhi Technological University, for the valuable support and guidance he provided in making this major project. It is my pleasure to record my sincere thanks to my respected guide for her constructive criticism and insight without which the project would not have shaped as it has.

I humbly extend my words of gratitude to other faculty members and my colleagues of this department for providing their valuable help and time whenever it was required.

Shraddha Chaudhary
Roll No. 13/VLSI/10
M.Tech (VLSI and Embedded systems)
E-mail: chaudhary.shraddha18@gmail.com

Abstract:

Transportation engineering has evolved into a broad multidisciplinary field during the last few decades. This multidisciplinary nature of the profession has become more profound and visible since the advent of the Intelligent Transportation Systems (ITS) early last decade. The remarkable advances in computers, communications, electronics, control, and other related technologies have found important applications in the transportation system. This thesis models one of the most important aspect of ITS i.e. prediction of the road traffic. In this thesis classification of the road traffic is done using Gaussian Mixture Model-Expectation Maximization (GMM-EM) and Prediction of road traffic is done using Hidden Markov model (HMM) . The proposed approach is neither based on tracking nor on vehicle detection hence, this approach has given new heights to the traffic monitoring systems. Apart from this, this method does not assume or draws analogies of traffic moving as particles, neither does it impose restriction on road conditions or road tributaries and distributaries. The results so obtained are verified and validated using the traffic simulator software.

Table of Contents

List of Figures

List of Tables

CHAPTER: 1

INTRODUCTION:

Traffic monitoring/prediction is presented in this thesis. The activities on each road link are monitored and features are derived to identify the pattern. Then it is learnt, classified, predicted and communicated to neighboring road links. The traffic model we developed extracts video based features and processes the same to classify the road condition as open, slight congestion, heavy congestion or traffic jam. It can also be used for predicting the traffic conditions along the road which is extended to find the best route to the destination. Currently, the dominant technology for this purpose is magnetic loop detectors, which are buried underneath highways to count vehicles passing over them. Video camera based systems present some advantages over the current technology. First, a much larger set of traffic parameters can be estimated in addition to vehicle counts and speeds. These include vehicle classifications, link travel times, lane changes, rapid accelerations or decelerations, queue lengths at urban intersections, etc. Second, cameras are less disruptive and less costly to install than loop detectors, which require digging up the road surface. Therefore, video cameras are becoming more popular in traffic monitoring and control systems.

Sensor nodes which can communicate to the nearby nodes are placed in all road links. A sensor node is equipped with, a camera with accessories to process the data and communication devices. The features (SIP and STIP) are derived from the video frames generated by the Camera. Spatial interest points are points (SIP) in spatial domain with significant variation in local intensities where as spatio-temporal interest points (STIP) are points in space time domain with significant variation in local intensities. It is observed that vehicles on a road generate SIP and the moving vehicles generate STIP. Hence, the number of SIP is indicative of number of vehicles on a road and ratio of STIP to number of SIP is suggestive of percentage of moving vehicles. These points are classified using GMM and prediction is done using HMM.

1.1 Motivation:

Sequential data arises in many areas of science and engineering. The data may either be a time series, generated by a dynamical system, or a sequence generated by a 1-dimensional spatial process, e.g., biosequences. One may be interested either in online analysis, where the data arrives in real-time, or in offline analysis, where all the data has already been collected. In online analysis, one common task is to predict future observations, given all the observations up to the present time. So we can say that in online learning although we have much complexity but that is worth learning because it gives us flavor of different types of scenarios which we could have not thought in the offline learning. More to it data that is obtained after online/real time learning is more exacting and efficient. Since we will generally be unsure about the future, we would like to compute a best guess. In addition, we might want to know how confident we are of this guess, so we can hedge our bets appropriately. Hence we will try to compute a probability distribution over the possible future observations. In this case, we would like to predict future outcomes as a function of our inputs with the help of transmission and emission probabilities matrix which play a major role in the HMM model. More to it HMM is a kind of simple dynamic Bayesian network (A dynamic Bayesian network (DBN) is a way to extend Bayes nets to model probability distributions over semi-infinite collections of random variables, Z1;Z2; : : :. Typically we will partition the variables into $Z_t = (U_t; X_t; Y_t)$ to represent the input, hidden and output variables of a state-space model. We only consider discrete-time stochastic processes, so we increase the index t by one every time a new observation arrives. (The observation could represent that something has changed (as in, making this a model of a discrete-event system.) Note that the term “dynamic” means we are modeling a dynamic system, not that the network changes over time.)Hence, this dynamism in the problem of predicting traffic led us to work on this project.

1.2 Related Work

Traffic modeling has gained significant interest among researchers and lot of work has been carried out already. Even then it is still a challenging work due to its nonlinear nature. The existing research works can be divided into two groups (i) Traffic classification and (ii) classification and prediction.

Most of the existing road classification research works are based on image segmentation and tracking vehicle, which cannot be used for on line learning due to the computational Complexity [10] [5] [14] and [7]. Peng Cheng and et.al. [4] Developed a particle filter based traffic estimation utilizing hand off data. In this case every vehicle should contain a cell phone in it, which is not practical always. Ren C Boel and et. al. have developed a hybrid stochastic traffic model [2]. The whole network is divided into sections and the traffic is predicted by using sending and receiving functions. Later they have developed a particle filter based modeling which is faster and can even be used for online prediction. But the main disadvantage was to detect vehicle in every section which makes complex computation. X.Li and et. al. [9] in their work "A Hidden Markov Model framework for traffic event detection using video features" used DCT in spatial and temporal domain as features and used HMM for event detection. This method is computationally complex due to the need of image processing which makes online learning difficult. Y. Zou and et.al [18] have developed HMM based traffic incident detection. The method is location specific and also needs vehicle detection. In general traffic modeling is done with image processing and tracking. We are presenting a model which do not require tracking and hence suitable for online prediction. Yan Qi in his Ph.D. [12] thesis "Probabilistic Models For Short Term Traffic Conditions Prediction" compared the traffic prediction using HMM based model and one step stochastic model. He derived traffic features from embedded magnetic loops on the road. Embedding magnetic loop on the road will be a tedious effort. In general all the above methods belong to group (i) and were doing traffic modeling on an individual road link and not extended for real time navigation.

1.3 Thesis Layout:

The project is divided into 6 chapters. Each chapter deals with one component related to this project work. Chapter 1 being introduction to this project work gives us the brief introduction to this project work, there after chapter 2 deals with the history of traffic monitoring systems. This chapter shows how our approach is novel, practical and easily applicable as compared to previous approaches as given in section 1.2.

Chapter 3 deals with the detection of vehicles on the road. The significant development that we have made in the project as compared to previous methods is that without the use of tedious vehicle detection methods we have been able to judge the amount of traffic on the road. Chapter 4 deals with classification of the road traffic using GMM. Each road state can be divided as Open, Slight Congestion, Heavy Congestion and Stopped. Learnt data so obtained was bell shaped and hence GMM classification became an obvious choice. Thereafter expectation maximization algorithm is used in order to optimize the results. Then comes the most important part to this project i.e. chapter 5 prediction is one of the most tedious part in the real time applications because we have no idea that what would happen next minute, taking this as our motivation as well we have tried and predicted the future road traffic state on the basis of the hand labeled learned data and HMM. Then the states so predicted are verified and validated using the traffic simulator. Lastly, in Chapter 6 all the results and simulations are shown and discussed. We have verified the system in different weather conditions (rainy, night, Day and foggy). Moreover, verification of the results was tested successfully using Sychro Studio 8 traffic simulator. Future applications to this work have also been discussed in this work.

CHAPTER: 2

TRAFFIC MONITORING

The importance of road transportation systems to our daily lives can hardly be overstated. To facilitate monitoring and management of their complexities, sensors collecting traffic information (such as traffic volumes and speeds) are installed on many roads. Today, coverage is good for major highways in many metropolitan areas and traffic sensor deployment is rapidly increasing worldwide. Road networks exhibit strong interaction patterns among traffic variables and traffic is subject to stochastic fluctuations. The ability to adequately model, reason about and predict stochastic behavior is crucial to computational support of many traffic management tasks, such as traffic routing, congestion analysis and accident detection. The number of deaths and injuries resulting from traffic accidents has been rapidly increasing. Most accidents occur at intersections. Thus, it is important to monitor traffic events at intersections. Some successful monitoring systems employ non vision sensors for this purpose. Though effective, such non-vision sensors have rather limited scope of usage. One of the most important advantages of utilizing vision sensors is the ability to collect rich information such as parking vehicles, traffic jams, and traffic violations. Monitoring traffic using a vision oriented system in crowded urban areas is a challenging task.

To have an overview of the traffic flow through the city can be very advantageous when new roads in the city are built or reconstructed. Also traffic light wait times can be programmed more efficiently when long-term traffic flow through the city is known. Current approaches of monitoring traffic include manual counting of vehicles, or counting vehicles using magnetic loops on the road. The main drawback of these approaches, besides the fact that they are expensive, is that these systems only count. Vehicle classification, speed and location registration for individual vehicles and vehicle size, are properties that are not, or only limited, supported by these systems.

The traffic model we developed extracts video based features and processes the same to classify the road condition as open, slight congestion, heavy congestion or traffic jam. It can also be used for predicting the traffic conditions along the road which is extended to the best route to the destination. Currently, the dominant technology for this purpose is magnetic loop detectors, which are buried underneath highways to count vehicles passing over them.

Video camera based systems present some advantages over the current technology. First, a much larger set of traffic parameters can be estimate. Second, cameras are less disruptive and less costly to install than loop detectors, which require digging up the road surface. Therefore, video cameras are becoming more popular in traffic monitoring and control systems.

2.1 Traffic Monitoring System

Figure 2.1: Cross sectional view of the Road intersection from camera

We set up a video camera on the roof of a building beside an intersection, in order to observe traffic activities. Analog colored video images, obtained by the camera, are transferred to our laboratory through the NTT optical fiber line. Images are divided into two channels in our laboratory. Images in one channel are directly recorded into **SVHS;** those on the other channel are captured through SGI video capture board, and digitized in real time for composing a background image. Current accident detection is done off-line due to the limited performance of SGI-02, while composing background images is running all day long in real time. [7]

To detect vehicles effectively, it is necessary to estimate the motion of each vehicle. An image of 640x480 pixels is divided into 80x60 blocks, where each block consists of 8x8 pixels. The motion vector at each block is estimated by the block matching technique. Typically, about 10 blocks belong to a vehicle, while only two or three blocks belong to motorcycles. Here, we refer to vehicles and motorcycles as objects.

Difficulties in object tracking are as follows:

(1) **Occlusion effect:** When an object rounds a corner of **an** intersection, some of the blocks to the object are self-occluded, and motion vectors at those blocks cannot be estimated correctly. When objects overlap with each other from the view of the camera which looks down the intersection slantwise from the top of the building, a large scaled occlusion also occurs.

(2) **Size effect:** When an object rounds a corner, the number of the blocks to an object varies due to appearance and disappearance of surfaces of the object. The size of an object also depends on the position of the object.

2.2 Flow Graph of Traffic monitoring System:

To make sure a traffic system runs in real-time, the frame rate can be reduced in this stage by not processing each frame. Also a reduction of camera resolution is possible by rescaling each incoming image frame. Another important issue in this first stage is how to store the data of the

Image frames from the stream. Some algorithms only handle luminance intensity which requires only one scalar value for each pixel in the image. In this case only a “black and white” Video stream is needed. However, when using information of each color channel in colored frames, better subtraction results can be obtained, especially when objects in the scene are present in low-contrast areas. A modern camera equipped with a CCD sensor linearly transforms infinite-dimensional spectral color space to a three-dimensional RGB color space via red, green and blue color filters. [2][5]

This procedure was used previously, in order to overcome the complexities and time consumption a novel approach has been proposed which basically extracts the important features from the video and with the use of those features or feature vector, various traffic conditions are distinguished which helps in describing the state of the road. In this proposed novel approach we have used Sensor nodes which can communicate to the nearby nodes are placed in all road links. A sensor node is equipped with, a camera with accessories to process the data and communication devices. The features (SIP and STIP) are derived from the video frames generated by the camera. Spatial interest points are points (SIP) in spatial domain with significant variation in local intensities where as spatio-temporal interest points (STIP) are points in space time domain with significant variation in local intensities. It is observed that vehicles on a road generate SIP and the moving vehicles generate STIP. Hence, the number of

SIP is indicative of number of vehicles on a road and ratio of STIP to number of SIP is suggestive of percentage of moving vehicles. These points are classified using GMM and then traffic state

Prediction is carried out using HMM. Once the next state of each road link is available, we can obtain the optimal path by assigning weights (in accordance with the predicted state) to each Link. Traffic modeling has gained significant interest among researchers and lot of work has been carried out already. Even then it is still a challenging work due to its nonlinear nature [20]. The existing research works can be divided into two groups

(i) Traffic classification and

(ii) Classification and prediction.

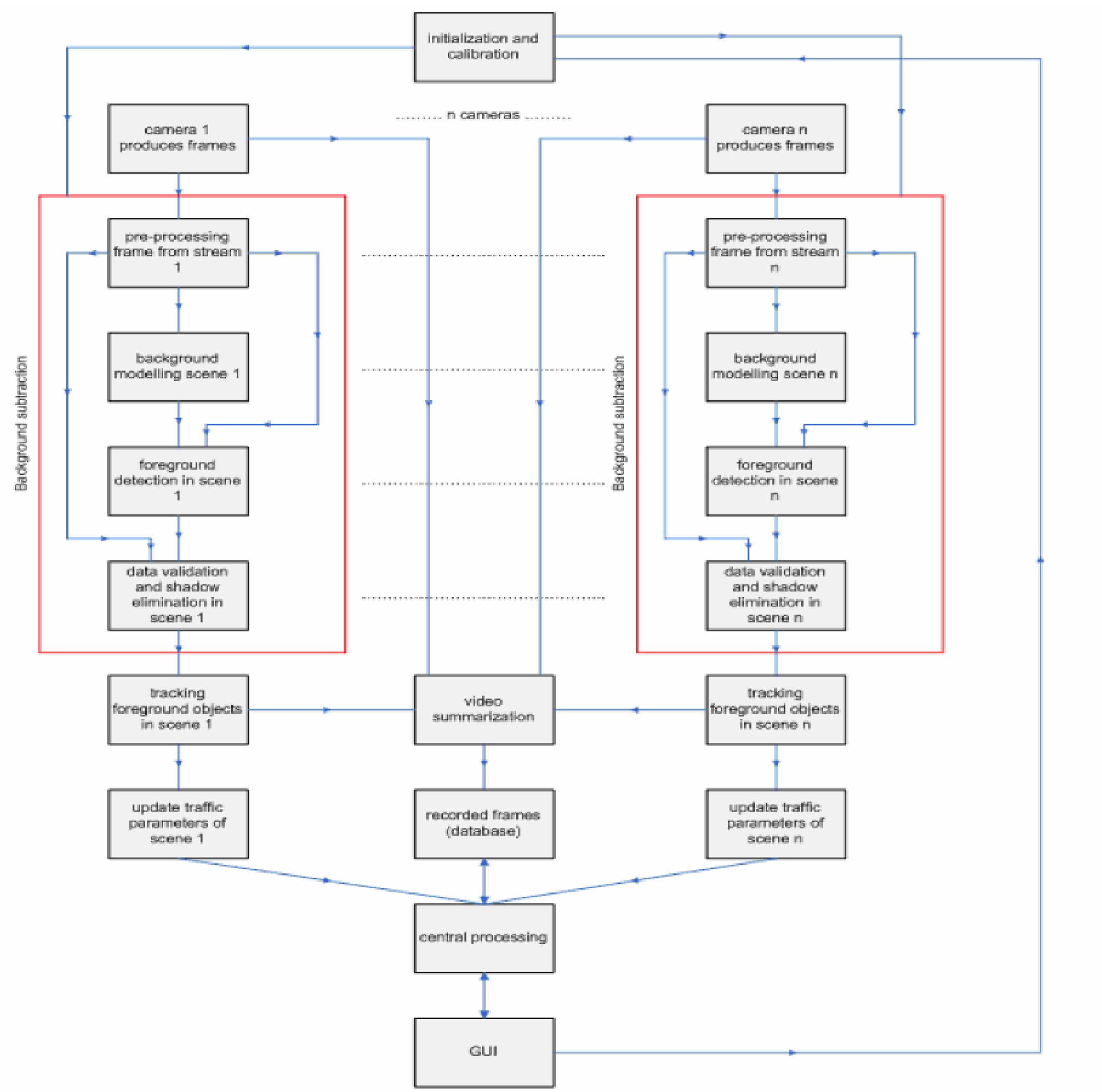


Figure 2.2: Flow Graph for Traffic Monitoring (Old Approach) [5]

2.3 Problem Formulation:

The main aim of the proposed work is to develop an autonomous distributed camera network for traffic prediction. A sample road network is shown in figure 2.3 R1 - R10 are different road links and C1 - C10 are cameras for recording the activities on the road links respectively. The video features of each road link will be extracted, processed, predicted and communicated to the nearby nodes periodically by each node. Each node will be sending the predicted states of the road and neighboring road links along with the ID of the respective road link for updating the current status. Traffic on any road can be completely defined by the number of moving vehicles and their average velocity. But these two features depend on each other. Therefore we classify the road states by comparing the no. of vehicles with the number of moving vehicles. If the no. of vehicles in a road link is η and the no. of moving vehicle is γ, we can classify the road state domain. The major classification considered is: Stopped (S), Heavy congestion (HC), Slight Traffic (ST), and Open (O) .We can extract SIP and STIP from the video frames recorded by the camera. The classification of the traffic state can be done based on the ratio of STIP to SIP as the ratio will give the indication of the state of the road. This is possible only when we get a dense feature set of correlated SIP and STIP, whereas the existing operators are providing sparse feature set of uncorrelated SIP and STIP. Hence we made a novel spatial interest point detector which provides dense feature set of correlated points by modifying the Harris corner detector.

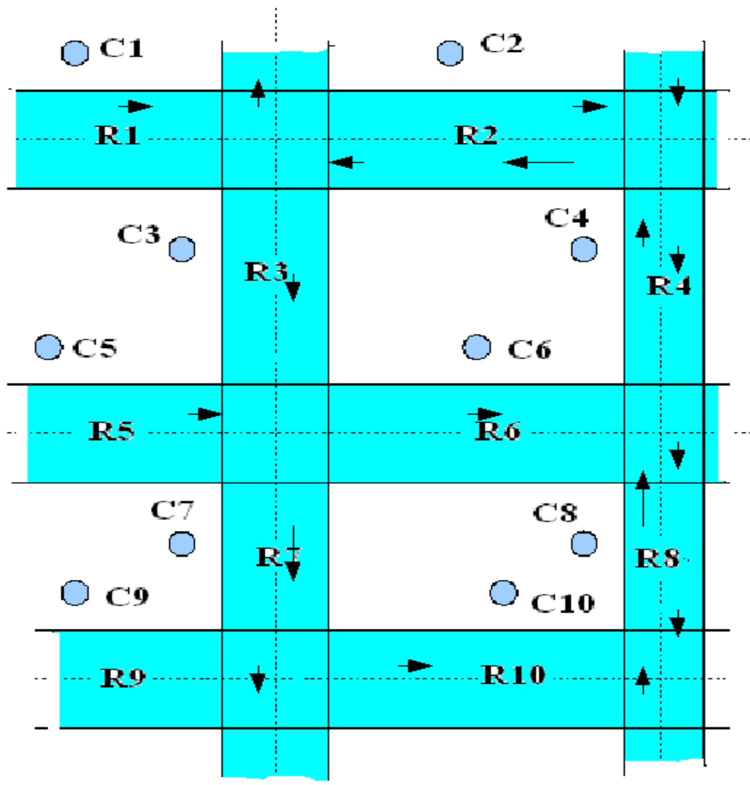


Figure 2.3: Road network

The parametric model of the saliency features distribution is learned by fitting a Gaussian Mixture Model (GMM) using the Expectation-Maximization (EM) algorithm on a hand labeled training data set. This approach is used because it is suitable for fast data processing with only three features, weight, mean and covariance.

CHAPTER-3

INTRODUTION TO SIP AND STIP:

Analyzing and interpreting video is a growing topic in computer vision and its applications. Video data contains information about changes in the environment and is highly important for many visual tasks including navigation, surveillance and video indexing. Basically by the term spatio: mean space i.e. f(x,y) where the given function describes about the given pixel of an image. Whereas temporal as the name itself refers is time, which means change of the space in accordance with time can be interpreted by the function: f(x,y,t).

Taking an example for the particular scenario, suppose we have a video, and in that we have an object that is performing certain action, so definitely if it's an video that action will vary in accordance with time more to it background will also keep on changing as the time is increased. so at this point comes the application or we can say importance of spatial and temporal interest points, which help in deducing the particular action, which is not constant hence is varying.

Image structures in video are not restricted to constant velocity and/or constant appearance over time. On the contrary, many interesting events in video are characterized by strong variations in the data along both the spatial and the temporal dimensions. Considering another example a scene with a person entering a room, applauding hand gestures, a car crash or a water splash; see also the illustrations in Figure.

More generally, points with non-constant motion correspond to accelerating local image structures that may correspond to accelerating objects in the world. Hence, such points can be expected to contain information about the forces acting in the physical environment and changing its structure.Local image features or interest points provide compact and abstract representations of patterns in an image. In this paper, we extend the notion of spatial interest points into the spatio-temporal domain and show how the resulting features often reflect interesting events that can be used for a compact representation of video data as well as for interpretation of spatio-temporal events.

To detect spatio-temporal events, we build on the idea of the Harris and Forester interest point operators and detect local structures in space-time where the image values have significant local variations in both space and time. We estimate the spatio-temporal extents of the detected events by maximizing a normalized spatio-temporal Laplacian operator over spatial and temporal scales. To represent the detected events, we then compute local, spatio-temporal, scale-invariant *N*-jets and classify each event with respect to its jet descriptor. [19]

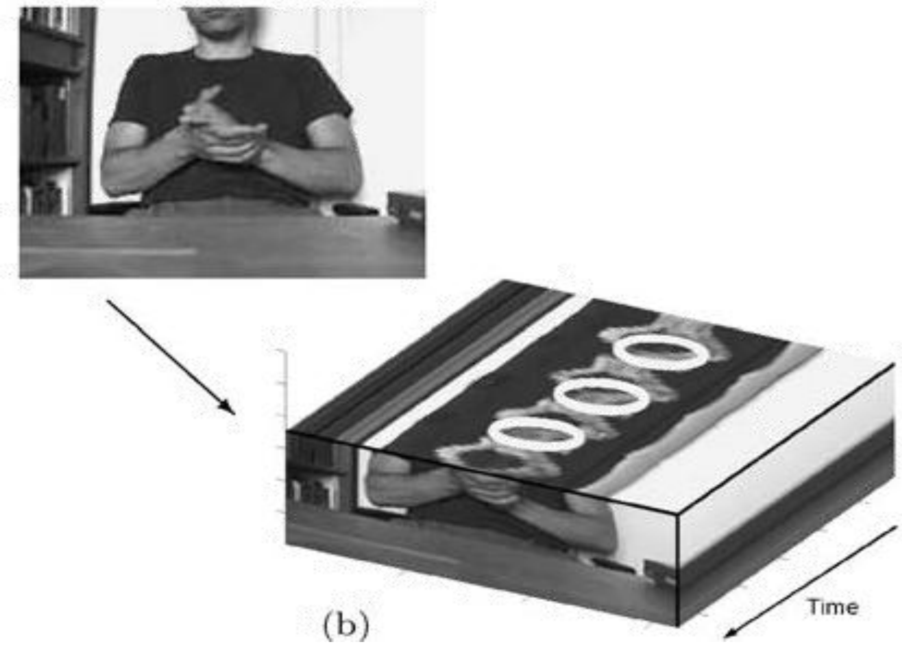


Figure 3.1: Example: If a person is applauding

3.1 Interest Points in Spatial Domain:

In the spatial domain, we can model an image f_{sp} : R^2 $|\rightarrow$ R by its linear scale-space representation (Witkin, 1983; Koenderink and van Doorn, 1992;

Lindeberg, 1994; Florack, 1997) L_{sp}:$R^2 \times R_+$ $|\rightarrow$ R

----------------(1)

Defined by the convolution of f^{sp} with Gaussian kernels of variance σ_l^2

------------------ *(2)*

The idea of the Harris interest point detector is to find spatial locations where f^{sp} has significant changes in both directions. For a given scale of observation σ^2_l, scale points can be found using a second moment matrix integrated over a Gaussian window with variance σ^2_l (Forester and Gulch, 1987; Begun et al., 1991; G°arding and Lindeberg, 1996):[19]

$$= g^{sp}(\cdot;\, \sigma_i^2) * \begin{pmatrix} (L_x^{sp})^2 & L_x^{sp} L_y^{sp} \\ L_x^{sp} L_y^{sp} & (L_y^{sp})^2 \end{pmatrix} \text{------------------ (3)}$$

Where '*' denotes the convolution operator, and $L^{sp}x$ and $L^{sp}\ y$ are Gaussian derivatives computed at local scale σ^2_l according to $L^{sp}x = \partial x\ (g^{sp}\ (\cdot;\ \sigma^2_l\ ) * f^{sp}(\cdot))$and $L^{sp}y = \partial y\ (g^{sp}(\cdot;\ \sigma^2 l\ ) * f^{sp}(\cdot))$. The second moment descriptor can be thought of as the covariance matrix of a two-dimensional distribution of image orientations in the local neighborhood of a point. Hence, the eigen values $\lambda 1$, $\lambda 2$, ($\lambda 1 \leq \lambda 2$) of μ^{sp} constitute descriptors of variations in f^{sp} along the two image directions. Specifically, two significantly large values of $\lambda 1$, $\lambda 2$ indicate the presence of an interest point. To detect such points, Harris and Stephens (1988) proposed to detect positive maxima of the corner function

----------------------- (4)

At the positions of the interest points, the ratio of the eigen values $\alpha = \lambda 2/\lambda 1$ has to be high. From (4) it follows that for positive local maxima of H^{sp}, the ratio α has to satisfy $k \leq \alpha/$ (1+α) 2. Hence, if we set $k = 0.25$, the positive maxima of H will only correspond to ideally isotropic interest points with $\alpha = 1$, i.e. $\lambda 1 = \lambda 2$. Lower values of k allow detecting interest points with more elongated shape, corresponding to higher values of α. A commonly used value of k in the literature is $k = 0.04$ corresponding to the detection of points with $\alpha<23$.

3.2 Interest Points in Spatio-Temporal Domain:

In this section, we develop an operator that responds to events in temporal image sequences at specific locations and with specific extents in space-time. The idea is to extend the notion of interest points in the spatial domain by requiring the image values in local spatio-temporal volumes to have large variations along both the spatial and the temporal directions. Points with such properties will correspond to spatial interest points with distinct locations in time corresponding to local spatio-temporal neighborhoods with non-constant motion.

To model a spatio-temporal image sequence, we use a function $f: R^2 \times R \rightarrow R$ and construct its linear scale-space representation $L: R^2 \times R \times R^2_+ \mid\rightarrow R$ by convolution of f with an anisotropic Gaussian kernel1 with independent spatial variance σ^2_l and temporal variance τ^2_l

-------------------------- (5)

Where the spatio –temporal separable Gaussian kernel is defined as:

--------------- (6)

Using a separate scale parameter for the temporal domain is essential, since the spatial and the temporal extents of events are in general independent.[23][24]

Similar to the spatial domain, we consider a spatiotemporal second-moment matrix, which is a 3-by-3 matrix composed of first order spatial and temporal derivatives averaged using a Gaussian weighting function:

)

$$\mu = g(\cdot;\, \sigma_i^2, \tau_i^2) * \begin{pmatrix} L_x^2 & L_xL_y & L_xL_t \\ L_xL_y & L_y^2 & L_yL_t \\ L_xL_t & L_yL_t & L_t^2 \end{pmatrix}$$ ------------------------------------ (7)

Where we here relate the integration scales σ^2_i and τ^2i to the local scales σ^2_l and τ^2_l according to σ^2_i= sσ^2l and τ^2_i= sτ^2_l. The first-order derivatives are defined as:

--------------------- (8)

To detect interest points, we search for regions in f having significant eigen values $\lambda1$, $\lambda2$, $\lambda3$ of μ. Among different approaches to find such regions, we propose here to extend the Harris corner function (4) defined for the spatial domain into the spatio-temporal domain by combining the determinant and the trace of μ as follows:

----------- (9)

To show how positive local maxima of H correspond to points with high values of $\lambda1$, $\lambda2$, $\lambda3$ ($\lambda1 \leq \lambda2 \leq \lambda3$), we define the ratios $\alpha = \lambda2/\lambda1$ and $\beta = \lambda3/\lambda1$ and re-write H as

------------ (**10**)

From the requirement $H \geq 0$, we get $k \leq \alpha\beta/(1+\alpha+\beta)3$ and it follows that k assumes its maximum possible value $k = 1/27$ when $\alpha = \beta = 1$. For sufficiently large values of k, positive local maxima of H correspond to points with high variation of the image values along both the spatial and the temporal directions. In particular, if we set the maximum value of α, β to 23 as in the spatial domain, the value of k to be used in H (8) will then be k ≈ 0.005. Thus, spatio-temporal interest points of f can be found by detecting local positive spatio-temporal maxima in H.

3.3 Experimental results:

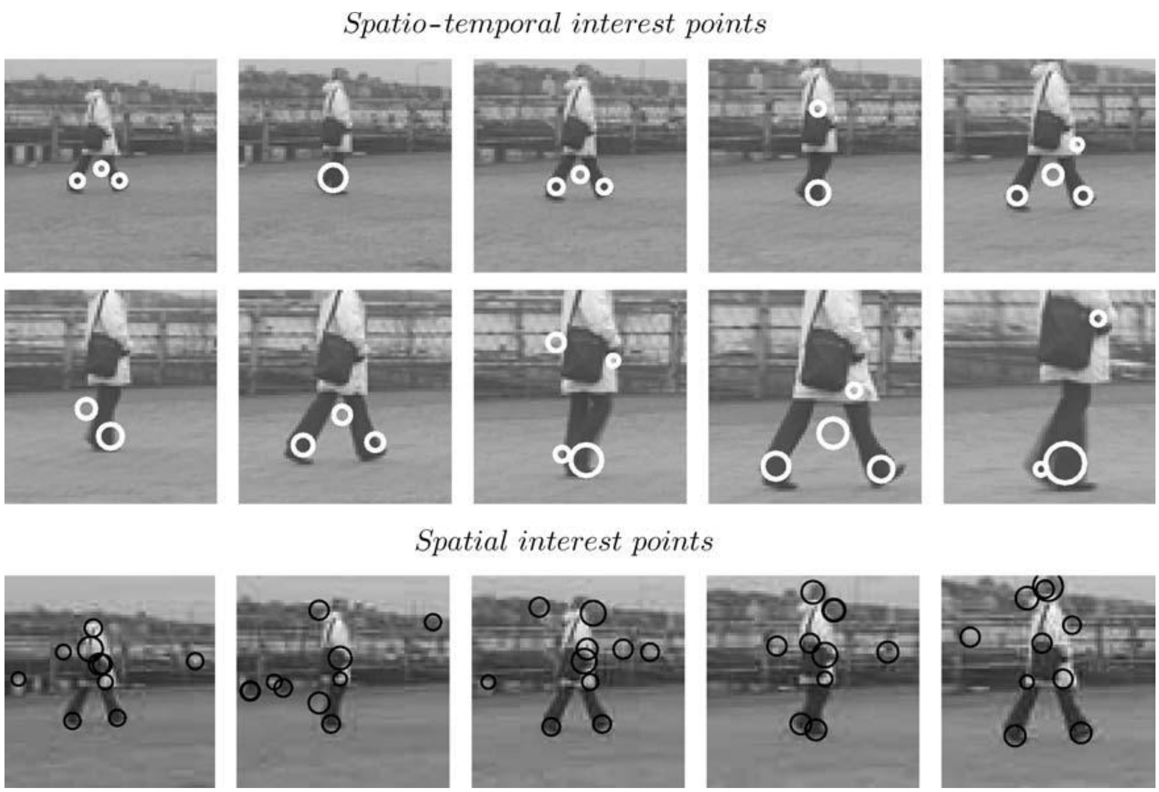


Figure 3.2: Results of spatio-temporal interest point detection for a zoom-in sequence of a walking person. The spatial scale of the detected points (corresponding to the size of circles)

matches the increasing spatial extent of the image structures and verifies the invariance of the interest points with respect to changes in spatial scale. Bottom: Pure spatial interest point detector (here, Harris-Laplace, Mikolajczyk and Schmid, 2001) selects both moving and stationary points in the image sequence[19]

CHAPTER: 4

CLASSIFICATION AND EXPECTATION MAXIMIZATION:

When something is measured, a piece of data is acquired. Naturally measurements are never exact, and at least some kind of noise is always expected in the data. Getting the data is not in the focus but the problem is to understand what the data means. That is, we need to classify the data. For example, if the vibration of an electric motor is measured, how can one tell from the results if the bearings have gone bad? The data is usually so vague or complex that we cannot just say,”If it is like this, then the case is that.” We need to develop a classifier that takes the data, examines it and gives a meaning to it according to experience and knowledge of the phenomena.

Classification is a very common task in information processing, and it is not generally easy. Some of the application areas are industrial quality control, object recognition from a visual image, and separating different voices from hum of voices. While these tasks might be easy for human beings they are very hard for machines. Even with the simplest cases there are noise and distortions affecting the measurement results and making the classification task nontrivial.

Classifier algorithms do not usually work well with raw data, such as a huge array of numbers representing a digital image. Data have to be preprocessed to extract few pieces of valuable information, called features. Features are represented as a feature vector where the dimension of a vector is the number of scalar components of different features. Feature extraction is very important for achieving good classification results and it is typically application specific. Usually each known sample, i.e., feature vector, is labeled to belong to a known class.

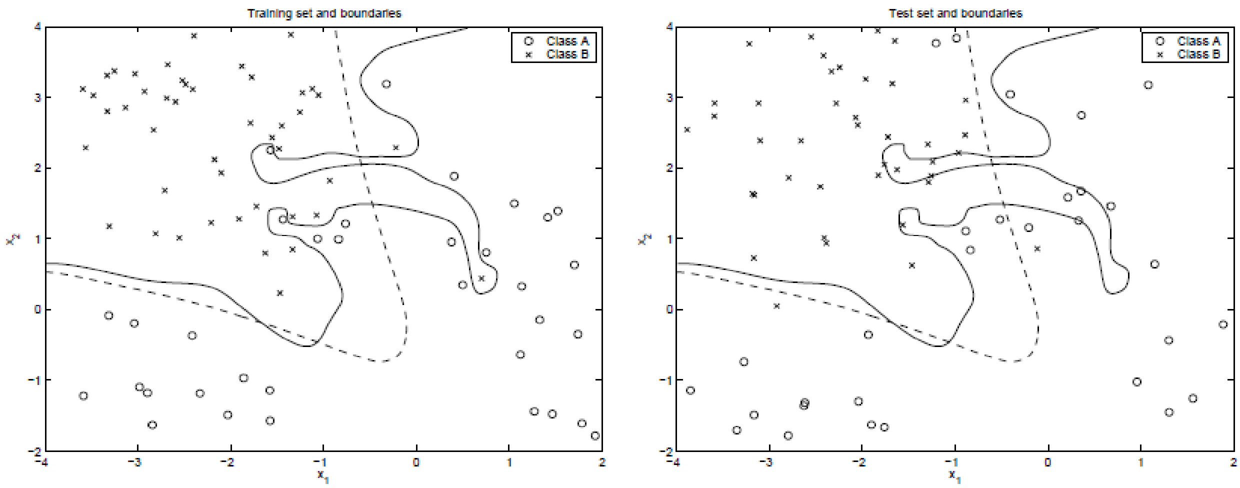
Feature vectors with class labels can be used to estimate a model describing a class, provided that there are enough good samples available. Some assumptions have to be made about the structure of the estimating model because totally arbitrary models are difficult to handle. For example in the Bayesian classification it can be assumed that a class can be represented in feature space with a Gaussian probability density function (PDF). The classification of unknown samples is based on estimated class representations in a feature space.

4.1 Classifier:

Classifier is an algorithm with features as input and concludes what it means based on information that is encoded into the classifier algorithm and its parameters. The output is usually a label, but it can contain also confidence values. Knowledge of a classification task can be incorporated into a classifier by selecting an appropriate classifier type, for example a neural network, a distance transform or the Bayesian classifier. Knowledge is also required to determine a suitable inner structure for the classifier, for example the number of neurons and layers in a neural network classifier. For the Bayesian classifier the probability density models, or functions, need to be selected. The complexity of a classifier is determined mostly by these choices. Classifier complexity is a tradeoff between representational power and generality for a task. A simple classifier may not be able to learn or represent classes well which yields poor accuracy.

An overly complex classifier can lead to a situation shown in Figure 4.1: an over fitted classifier can classify the training data 100% correct, but when a different data set from the same task is presented, the accuracy can be poor. Therefore the training data is usually divided into two disjoint sets: an actual training set and a test set, so that the classifier performance can be estimated objectively.

A classifier can have numerous parameters that have to be adjusted according to the task. This is called training or learning. In supervised learning the training samples are labeled, and the training algorithm tries to minimize the classification error of the training set. In unsupervised learning or clustering the training samples are not labeled but the training algorithm tries to find clusters and form classes. In reinforcement learning the training samples are also not labeled, but the training algorithm uses feedback saying if it classifies a sample correctly or not



(a) Training data plot. (b) Test data plot.

Figure 4.1: Too complex decision boundary (solid line) can separate the training set without any errors (Fig. (a)), but it can make many mistakes with another data set (Fig. (b)). Dashed line is a more general boundary that leads to lower average classification error. [25]

4.2 Gaussian Mixture Probability Density Function:

The Gaussian probability density function in one dimension is a bell shaped curve defined by two parameters, mean μ and variance σ^2. In the D-dimensional space it is defined in a matrix form as

-------------- (11)

Where μ is the mean vector and Σ the covariance matrix. In Figure 4.2 is shown an example of 2-dimensional Gaussian PDF. Equi probability surfaces of a Gaussian are μ -centered hyper ellipsoids.

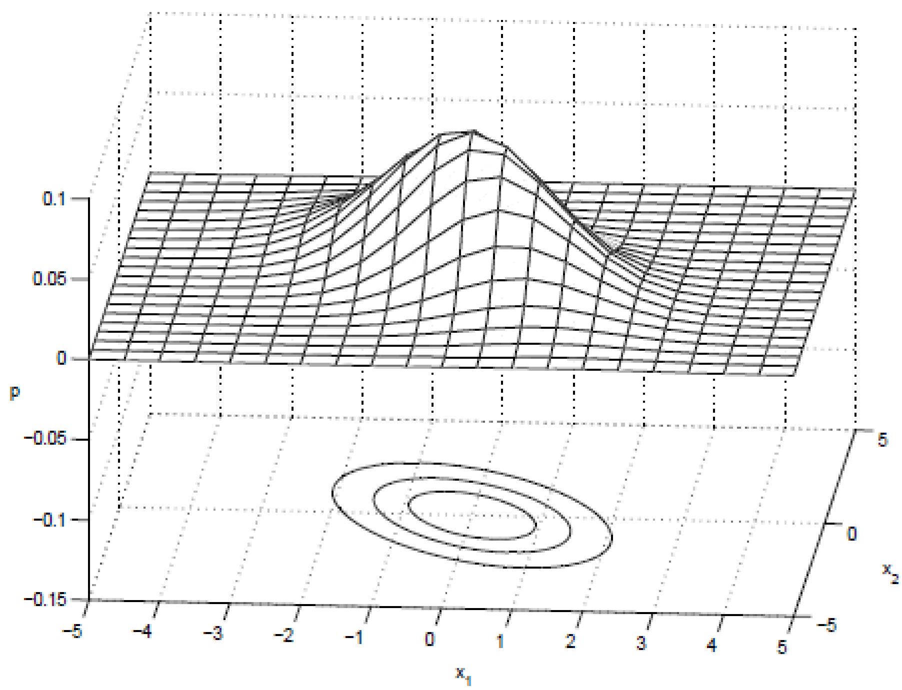


Figure 4.2: An example surface of 2-dimensional Gaussian Pdf .

The Gaussian distribution is usually quite good approximation for a class model shape in a suitably selected feature space. It is a mathematically sound function and extends easily to multiple dimensions. In the Gaussian distribution lies an assumption that the class model is truly a model of one basic class. If the actual model, the actual probability density function, is

multimodal, it fails. For example, if we are searching for different face parts from a picture and there are several basic types of eyes, because of people from different races perhaps, the single Gaussian approximation would describe a wide mixture of all eye types, including patterns that might not look like an eye at all.

Gaussian mixture model (GMM) is a mixture of several Gaussian distributions and can therefore represent different subclasses inside one class. The probability density function is defined as a weighted sum of Gaussians

-------------------------- (12)

where is the weight of the component c, 0 < αc < 1 for all components, and = 1.

The parameter list

------------------------- (13)

defines a particular Gaussian mixture probability density function. An example of Gaussian mixture model is shown in Figure 4.3.

Estimation of the Gaussian mixture parameters for one class can be considered as unsupervised learning of the case where samples are generated by individual components of the mixture

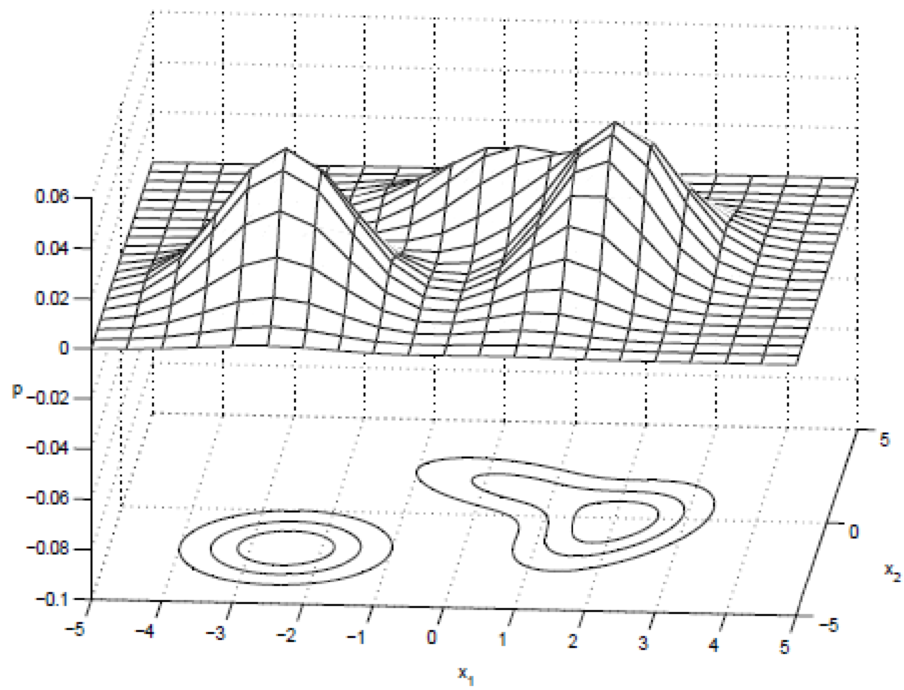


Figure 4.3: An example surface of a two-dimensional Gaussian mixture PDF with three Components( )

distribution and without the knowledge of which sample was generated by which component. Clustering usually tries to identify the exact components, but Gaussian mixtures can also be used as an approximation of an arbitrary distribution.

4.2.1 Complex Valued features:

Some applications utilize complex valued features, for example describing the amplitude and phase of a sinusoidal signal. These features can be handled in three different ways. First, a single real value can be calculated from each complex attribute, by probably losing some information. Second, the complex value can be divided to two real values, which means that the length of feature vectors is doubled. Third, it might be possible to use the complex arithmetic's directly.

The first approach is good in the sense that it reduces the dimensionality of a problem, which can lead to better performance even with small number of training data. If this is the case, it should have been done already in the feature extraction phase. On the other hand, combining two values into one can cause a loss of valuable information and the performance may degrade.

Dividing complex values into two parts, real and imaginary or modulus and argument, preserves all information, but increases the dimensionality of the problem. This can be a real problem with high dimensional complex valued data since the amount of training data may become insufficient or computation time increases too much. Using the complex values directly is something between the two previous cases. It introduces a constraint to the Gaussian distribution model and requires little modifications in classifier training algorithms and the Gaussian PDF. If the transpose operation in Eq. 11 is changed to conjugate transpose, the resulting formula

---- (14)

works also with complex valued μ and Σ, assuming the covariance matrix is still positive definite. The constraint affects the shape of the distribution. It forces the shape of the density function to be spherical on each complex plane, eliminating the covariance between the real and imaginary part of a variable and making the variances equal.

The differences between approaches for using complex valued data in the Gaussian PDF are presented in Table 1. The degree of freedom here refers to an unknown value and the numbers of such values in the D- dimensional case are shown in the table. With the covariance matrix Σ the symmetry and real valued diagonal constraints are taken into account.

Table 1: Comparison of degrees of freedom in a single multivariate Gaussian density function with different approaches for using complex data.[25]

	Degrees of Freedom	
Approach	μ	Σ
Convert to single real value	D	$\frac{1}{2}D^2 + \frac{1}{2}D$
Divide to two real values	$2D$	$2D^2 + D$
Use complex value	$2D$	D^2

4.3 Gaussian Mixture Density Estimation:

In construction of a Bayesian classifier the class-conditional probability density functions need to be determined. The initial model selection can be done for example by visualizing the training data, but the adjustment of the model parameters requires some measure of goodness, i.e., how well the distribution fits the observed data. Data likelihood is a such goodness value.

Assume that there is a set of independent samples drawn from a single distribution described by a probability density function , where is the PDF parameter list. The likelihood function:

----------- (15)

tells the likelihood of the data X given the distribution or, more specifically, given the distribution[21] ; parameters . The goal is to find that maximizes the likelihood:

$$= \quad \text{-------------------- (16)}$$

Usually the function is not normalized but the logarithm:

$$= \quad \text{------- (17)}$$

Called the log-likelihood function it is easier to handle.

Depending on , it is possible to find maximum analytically by setting the log likelihood function and solving for . It can be done for a Gaussian PDF, which leads to the intuitive estimates for a mean and variance. But usually the analytical approach is intractable. In practice an iterative method such as the expectation maximization (EM) algorithm is used. Maximizing the likelihood may in some cases lead to singular estimates, which is the fundamental problem of maximum likelihood methods with Gaussian mixture models.

4.4 Basic Expectation Maximization

The expectation maximization (EM) algorithm is an iterative method for calculating maximum likelihood distribution parameter estimates from incomplete data (some elements missing in some feature vectors). It can also be used to handle cases where an analytical approach for maximum likelihood estimation is infeasible, such as Gaussian mixtures with unknown and unrestricted covariance matrices and means.

Assume that each training sample contains known features and missing or unknown features. Mark all good features of all samples with X and all unknown features of all samples with Y. The expectation step (E-step) for the EM algorithm is to form the function

---------------- (18)

where is the previous estimate for the distribution parameters and is the variable for a new estimate describing the (full) distribution. L is the likelihood function. The function calculates the likelihood of the data, including the unknown feature Y marginalized with respect to the current estimate of the distribution described by . The maximization step (M-step) is to maximize Q(;) with respect to and set

-------------- (20)

The steps are repeated until a convergence criterion is met.

The EM algorithm starts from an initial guess for the distribution parameters and the log-likelihood is guaranteed to increase on each iteration until it converges. The convergence leads to a local or global maximum, but it can also lead to singular estimates, which is true particularly for Gaussian mixture distributions with arbitrary covariance matrices. [1]

The initialization is one of the problems of the EM algorithm. The selection of (partly) determines where the algorithm converges or hits the boundary of the parameter space producing singular, meaningless results. Some solutions use multiple random starts or a clustering algorithm for initialization. [16]

The application of the EM algorithm to Gaussian mixtures according to [16] goes as follows. The known data X is interpreted as incomplete data. The missing part Y is the knowledge of which

component produced each sample . For each there is a binary vector ,where , if the sample was produced by the component c, or zero otherwise. The complete data log-likelihood is

= ---- (21)

The E-step is to compute the conditional expectation of the complete data log-likelihood, the Q function, given X and the current estimate of the parameters. Since the complete data log-likelihood is linear with respect to the missing Y, the conditional expectation has simply to be computed and put it into .

Therefore

----------- (22)

where the elements of W are defined as

-------- (23)

The probability can be calculated with the Bayes law

------------------------------------ (24)

where is the a priori probability (of estimate) and is the a posteriori probability that after observing . In other words, is the probability that was produced by component c. [16]

Applying the M-step to the problem of estimating the distribution parameters for C-component Gaussian mixture with arbitrary covariance matrices, the resulting iteration formulas are as follows:

– --------------------- (25)

——————— --------------------- (26)

———————————————-- (27)

The interpretation of the Esq. 25-27 is actually quite intuitive. The weight of a component is the portion of samples belonging to that component. It is computed by approximating the component-conditional PDF with the previous parameter estimates and taking the posterior probability of each sample point belonging to the component c (Eq. 17).The component mean and covariance matrix are estimated in the same way. The samples are weighted with their probabilities of belonging to the component, and then the sample mean and sample covariance matrix are computed.It is worthwhile to note that so far the number of components C was assumed to be known. Clustering techniques try to find the true clusters and components from a training set, but our task of training a classifier only needs a good enough approximation of the distribution of each class. Therefore, C does not need to be guessed accurately, it is just a parameter defining the complexity of the approximating distribution. Too small C prevents the classifier from learning the sample distributions well enough and too large C may lead to an over fitted classifier. More importantly, too large C will definitely lead to singularities when the amount of training data becomes insufficient.

4.5 Application of GMM classification and expectation maximization in the project work:

In this project work, the features (SIP and STIP) are derived from the video frames generated by the camera. Spatial interest points are points (SIP) in spatial domain with significant variation in local intensities where as spatio-temporal interest points (STIP) are points in space time domain with significant variation in local intensities. It is observed that vehicles on a road generate SIP and the moving vehicles generate STIP. Hence, the number of SIP is indicative of number of vehicles on a road and ratio of STIP to number of SIP is suggestive of percentage of moving vehicles. These points are classified using GMM.

Traffic on any road can be completely defined by the number of moving vehicles and their average velocity. But these two features depend on each other. Therefore we classify the road states by comparing the no. of vehicles with the number of moving vehicles. If the no. of vehicles in a road link is and the no. of moving vehicle is , we can classify the road state ηγdomain. The major classification considered is: Stopped (S), Heavy congestion (HC), Mild congestion (MC), Slight Traffic (ST), and Open (O) We can extract SIP and STIP from the video frames recorded by the camera. The classification of the traffic state can be done based on the ratio of STIP to SIP as the ratio will give the indication of the state of the road. This is possible only when we get a dense feature set of correlated SIP and STIP, whereas the existing operators are providing sparse feature set of uncorrelated SIP and STIP. Hence we made a novel spatial interest point detector which provides dense feature set of correlated points by modifying the Harris corner detector.

The parametric model of the saliency features distribution is learned by fitting a Gaussian Mixture Model (GMM) using the Expectation-Maximization (EM) algorithm on a hand labeled training data set. This approach is used because it is suitable for fast data processing with only three features, weight, mean and covariance.

Using the spatial (η_{sp}) and spatio-temporal interest (γ_{pt}) points we can classify the road state by using Gaussian mixture model .Let the feature vector be S = ($\eta_{sp,}$ γ_{pt}) , no. of classes be k and no. of Gaussian mixtures be n.The conditional probability that belongs to K_{th} class is given by:

----- (29)

After simulating this formula and maximizing the expectation, we got 4 four clusters which tells us about the 4 classes that we have specified with different mean and variance values.

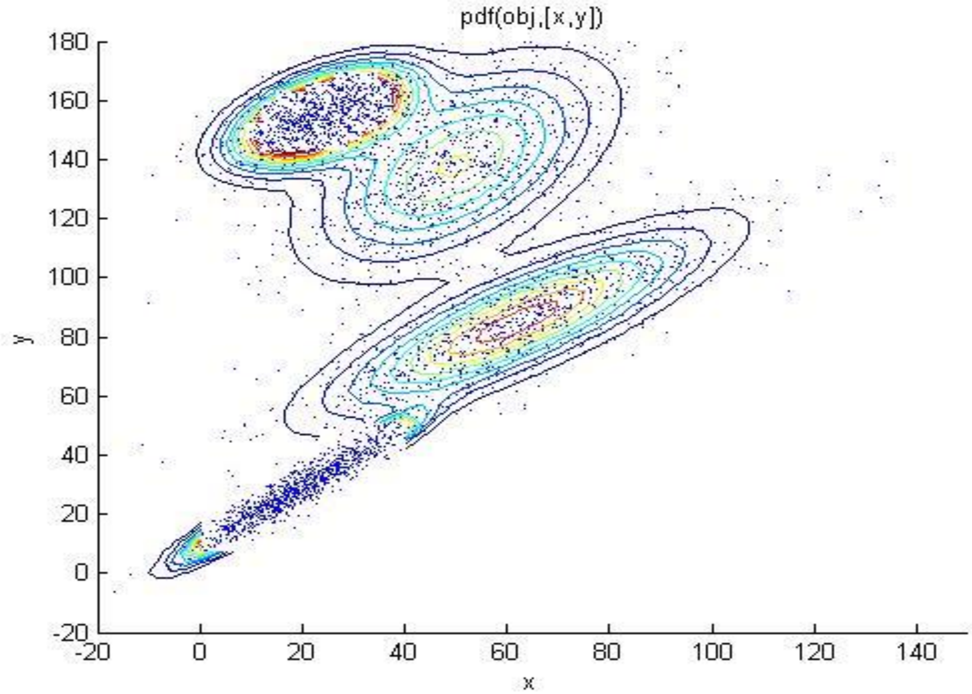


Figure 4.4: Representation shows 4 clusters each representing one of the road state.

CHAPTER: 5

PREDICTION USING: HMM

In this project we have used hidden Markov model for the traffic prediction. Traffic prediction is a real time process and hence requires system to be self evolving in order to reduce the complexity. Since HMM is a self evolving model sometimes it is considered as the simple dynamic Bayesian Network. In real-world processes generally produce observable outputs which can be characterized as signals. The signals can be discrete in nature (e.g., quantized vectors from code book, characters from a finite table etc.)Or continuous in nature (e.g., temperature measurements, speech samples, music etc.)The signal source can be stationary i.e. its statistical properties do not vary with time or non-stationary i.e. the signal properties vary with time.

A problem of fundamental interest is characterizing such real –world signals in terms of signal models. There are several reasons why one is interested in applying signal models. First of all, a signal model can provide the basis for a theoretical description of a signal processing system which can be used to process the signal so as to provide a desired output. For example if we are interested in enhancing speech signal corrupted by noise and transmission distortion, we can use the signal model to design a system which will optimally remove the noise and undo the transmission distortion. A second reason why signal models are important is that they are potentially capable of letting us learn a deal about the signal source available. This property is especially important when the cost of getting signals from the actual source is high. In this case with good signal model we can simulate the source and learn as much as possible via simulations. Finally, the most important is that they work extremely well in practice and enable us to realize important practical systems –e.g., prediction systems, recognition systems, identification systems, etc., in very efficient manner.

5.1: REPRESENTATION:

Before describing the Hidden Markov Model, it is necessary to describe its foundation, the Markov process. In any pattern there is usually sufficient structure to influence the probability of the next event. For example, in the English language, the probability of detecting the letter u depends very much on whether the letter q was last detected, since u almost always follows q. A stochastic process is called a jth-order Markov process if the conditional probability density of the current event, given all past and present events, depends only on the most recent events.

A *Hidden Markov Model* (denoted l) is a doubly stochastic process where each state generates (emits) an observation. We will use X_t to denote the hidden state and Y_t to denote the observation. If there are K possible states, then X_t {1,……,K}. Y_t might be a discrete symbol, Y_t {1,……,L}, or a feature-vector, Y_t $R^{L.}$ The parameters of the model are the initial state distribution, (i) = P(X1 = i), the transition model, A(I, j) = $P(X_t = j | X_{t-1} = i)$, and the observation model $P(Y_t | X_t)$. Represents a multinomial distribution. The transition model is usually characterized by a conditional multinomial distribution: , where A is a stochastic matrix (each row sums to one). The transition matrix A if often sparse; the structure of the matrix is often depicted graphically, as in Figure 5.1 which depicts a left-to-right transition matrix. (This means low numbered states can only make transitions to higher numbered states or to themselves.)

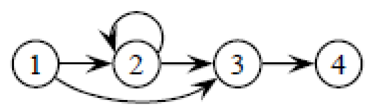


Figure 5.1 : A left-to-right state transition diagram for a 4-state HMM. Nodes represent states, and arrows represent allowable transitions, i.e., transitions with non-zero probability. The self-loop on state 2 means $P(X_t = 2 | X_{t-1} = 2) = A(2, 2) > 0$.

If the observations are discrete symbols, we can represent the observation model as a matrix:

$$B(i, k) = P(Y_t = k | X_t = i).$$

If the observations are vectors in IR^L, it is common to represent $P(Y_t | X_t)$ as a Gaussian:

$$P(Y_t = y | Xt = i) = N(y; \mu_i , \Sigma_i)$$

where $N(y; \mu, \Sigma)$ is the Gaussian density with mean μ and covariance Σ evaluated at y:

$$N(y\ ;\ \mu_i\ ,\ \Sigma_i) = \qquad \text{------------ (30)}$$

A more flexible representation is a mixture of M Gaussians:

$$P(\ Yt = y\ |\ Xt = i) =$$

Where, M_t is a hidden variable that specifies which mixture component to use, and $P(M_t = m \mid X_t = i) = C(i\ ,\ m)$ is the conditional prior weight of each mixture component. For example, one mixture component might be a Gaussian centred at the expected output for state i and with a narrow variance, and the second component might be a Gaussian with zero mean and a very broad variance; the latter approximates a uniform distribution, and can account for outliers, making the model more robust. In speech recognition, it is usual to assume that the parameters are stationary or time-invariant, i.e., that the transition and observation models are shared (tied) across time slices.[13] This allows the model to be applied to sequences of arbitrary length. In bio sequence analysis, it is common to use position-dependent observation models, $B_t(i\ ,\ k) = P(Y_t = k \mid X_t = i)$, since certain positions have special meanings. These models can only handle fixed-length sequences. However, by adding a background state with a position invariant distribution, the models can be extended to handle varying length sequences. In this thesis, we will usually assume time/position-invariant parameters, but this is mostly for notational simplicity.

5.2 LEARNING:

Learning phase in HMM is done with the help of maximum likelihood estimation parameter using EM(Baum-Welch) algorithm.[12]

If we could observer $_{1:T}$, learning would be easy For instance, the ML estimate of the transition matrix could be computed by normalizing the matrix of co-occurrences (counts):

———--------------------- (31) (i)

Where

= = ---- (31) (ii)

and is a binary indicator that is 1 if event occurs and is 0 otherwise. Hence is the number of transitions transition in a given sequence. (We have assumed there is a single training sequence for notational simplicity. If we have more than one sequence, we simply sum the counts across sequences.) Similarly, We can estimate | .

5.2.1 HMM in traffic estimation:

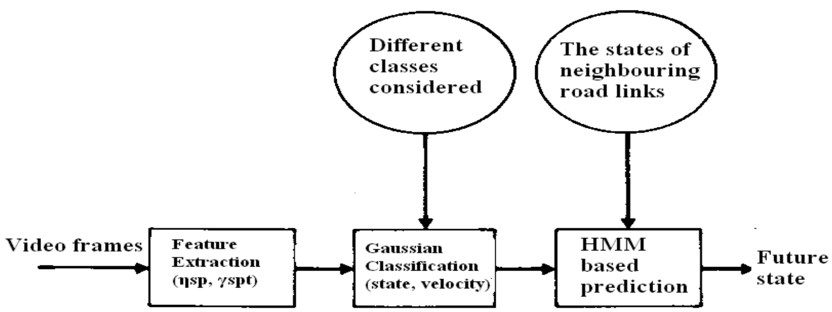


Figure 5.2: Traffic Prediction Model

Each state is characterized by two probability distributions: the transition distribution over states and the emission distribution over the output symbols. At each time step the source is in one state (ZK), and after emitting an output symbol according to the emission distribution of the current state (XK), the source jumps to a next state (ZK+1) according to the transition distribution of its current state (XK). Since the activity of the source is observed indirectly, through the sequence of output symbols, and the sequence of states is not directly observable, the states are said to be hidden. Since traffic event is a continuous process and the profile of the probability density function in one state is a combination of several Gaussian curves, we used HMM for Predicting the future state. HMM Topology used for continuous process is shown in Figure5.2 Let R1;R2;R3 be the three neighboring road links and RM is main road, linked together.[20]

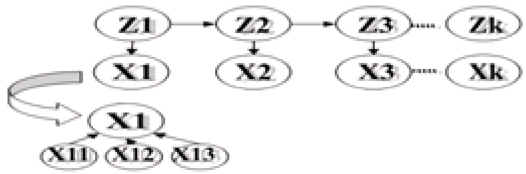


Figure 5.2.: State diagram of Road traffic prediction model

Xn1;Xn2;Xn3 and Zn are the traffic states of the road links respectively at nth time. For learning the 2 dimensional relation (spatial and temporal) of the traffic states, we use Hidden Markov Model with an assumption that for very short interval of time (refresh time of our system), the traffic condition at a given road link changes only due to the traffic conditions on neighboring road links. An elaborate list of sequence is created for the calculation of Transition and Emission Matrix. For prediction of Future state, we calculated the posterior state probabilities, of the sequence from a hidden Markov model. The posterior state probabilities are the conditional probabilities of being at state k at step I, given the observed sequence of symbols. Hence, transition, emission matrix and a particular sequence were used as input and we got the posterior state probability for that given sequence. Posterior state probability was calculated with help of the forward backward algorithm, which is given below.

---- (32)

Where p (Z_k | X) is posterior state probability, p(Z_k , X) is joint probability, p($X_{k+1:n}$ | Z_k) is backward part of algorithm and p(Z_k | $X_{1:k}$) is forward part of algorithm.

Forward part of algorithm is:

---------------- (32) (i)

Backward part of algorithm is:

--------------- (32) (ii)

Where p(X_k | Z_k) is Emission Probability , p(Z_k | Z_{k-1}) is Transition Probability , p(Z_{k-1} | $X_{1:k-1}$) is probability one step backward to p(Z_k | $X_{1:k}$) and p($X_{k+2:n}$ | Z_{k+1}) is probability one step forward to p($X_{k+1:n}$ | Z_k) .

Transition Matrix is given by:

------------------ (33)

5.2.2 Learning in traffic estimation:

We take the learnt GMMs as our class definitions, or class distribution initially. Then the road is kept under observation by the system for sufficient time to generate data, for learning transitions and optimizing state distributions. The temporal flow of states on the road is recorded while it is under observation using state definitions described earlier. The data so generated is used to learn HMM parameters through Balm - Welch algorithm. A compact description of the learning procedure is defined below:

- The initial state distributions represented as B are given by GMMS .
- These state distributions are used to observe and record the temporal flow of observed features on a road, using cameras and features described. Let this data or flow of features be represented by O
- The transition matrix for state distributions is initialized randomly and is represented by A.
- The initial state distribution is again given by GMMs learnt previously and is represented by .
- = arg max(P(O/)) is found by Baum-Welch algorithm, where . The , so obtained has tuned parameters as per the information available.

The same process can be followed periodically, making the system defined as self evolving and changing as the road conditions change. The transition matrix and state distributions are hence learnt on periodically making the system adaptive.

CHAPTER: 6

RESULTS, SIMULATIONS AND FUTURE SCOPE

6.1 Validation of states and feature selection at Rajouri Garden.

We evaluated the method suggested using the data collected from real traffic scenes. All testing clips are hand-labeled to make a comparison with a ground truth. The training video data is chosen such that there is no overlap with the testing data. The red points indicate spatial interest points and the blue points indicate spatial temporal interest points generated using the STIP operator. Interestingly, as the traffic state changes from open to heavy congestion, the number of spatial interest points become more as compared to number of corresponding spatial temporal interest points, hence further validating our feature selection .It is visible that all of the existing traffic states are successfully detected.

Figure 6.1: Each Frame illustrates a road traffic state, spatial and spatial temporal points in red and blues respectively.

6.2 Validation of states and feature vector for various ambient Conditions.

After studying and verifying the results in different traffic conditions, robustness of this algorithm and the feature vector was tested for the different ambient conditions .In India as we know weather Conditions play a major role and they are not steady throughout the year it was like an important goal for the fulfillment of the aim of our project. So the different ambient conditions studied are as follows:

when we are dealing with the real time traffic scenarios there could be times when the weather is rainy or in winters there could be the chance that there is fog , so for those conditions our system has evolved quitewell.

In the figure given below we can see that SIP and STIP points are detected well in the foggy weather condition. The thing that should be noted here is that very scarce or rather no light because of traffic lights. The SIP and STIP points were detected even in the absence of great lighting conditions

Fig 6.2: Detection of SIP and STIP points in the Foggy condition.

In the figure 6.3 given below, we can see that that the system was very equipped to judge the traffic conditions in the night conditions. As stated above, in the same way SIP and STIP points are detected. Hence, we can conclude that the system will not only work in night conditions but also on those roads where there is no proper arrangements of lights.

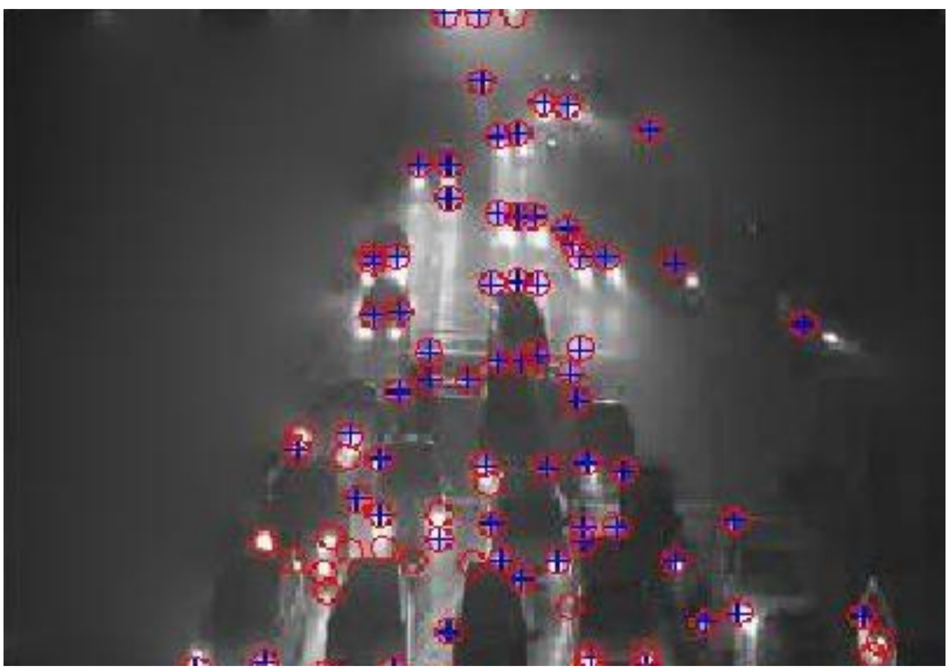


Figure 6.3: Detection of SIP and STIP points in Night Condition

Lastly, another weather condition could be the rainy condition. In the figure given below, we can see that system worked very well in this condition. As we could see that edges on the vehicles were easily detected. Rain drops, being transparent, don't have much of a role to play in the gray scale image that we were working on.

Figure 6.4: Detection of SIP and STIP points in the Rainy condition

6.2 Validating GMM as classifier using Confusion Matrix:

Apart from checking the robustness of the system in different traffic condition, we also checked our algorithm in different traffic condition and we came up with better confusion matrix. Moreover, we are also including the part of GMM where we have scattered five thousand points of each class. In the figure shown below, we have shown a plotting of SIP and STIP points on the graph. As we could see, there are 4 different clusters which have formed. Here Number of SIP points are on Y-axis and Number of STIP points are on X-axis. Slight pondering on the matter can get us a simple idea that in case of high traffic, number of SIP points will be high and number of STIP points will be low. In the given diagram, it is very evident.

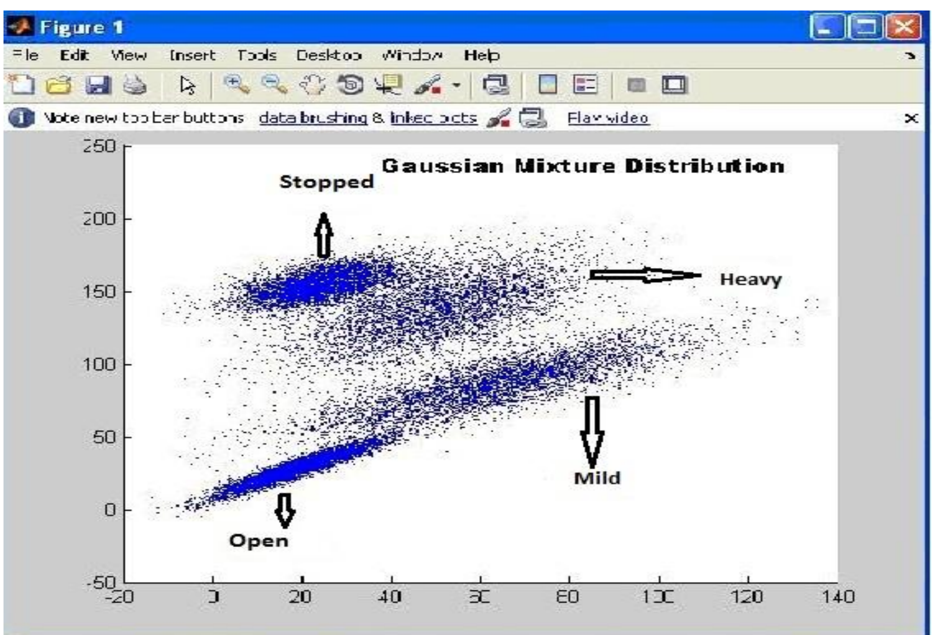


Figure 6.5: Graph showing scattering of SIP and STIP points

Confusion matrix is built to verify the result in actual conditions. Here the rows represent the actual condition and the columns show what we got through our algorithm. Hence for example a 98% in the first row of first column shows that when the actual condition was open then our algorithm got it right approx 98% of times. Hence a approx 1% in 1st column of 2nd row shows that the algorithm wrongly showed an open road condition as slightly mild on 1% of the total occurrences of the open road condition.

Table 2: (New) In comparison to previously published results[20] at Rajouri garden Road, Delhi

	open	**Slight Congestion**	**Heavy Congestion**	**Stopped**
Open	97.48	4.83	0	0
Slight Congestion	1.89	94.84	0.71	0
Heavy Congestion	0.63	0.23	96.82	1.33
Stopped	0	0	2.47	98.67

Table 3: (old) Confusion Matrix at the Rajouri garden Road, Delhi[20]

	Open	**Slight Congestion**	**Mild Congestion**	**Heavy Congestion**
Open	86.11521	3.148855	12.64479	3.374447
Slight Congestion	2.215657	81.10687	0.22522	0.698162
Mild Congestion	5.317578	9.446565	73.9704	4.21221
Heavy congestion	6.351551	6.29771	13.15959	91.71515

From the above Table 2 and Table 3 it is quite visible that result has been improved and new confusion matrix is better in comparison to the old one.

6.4 HMM Results:

The results that we get from the simulator are in tune with the results that we had after applying Hidden Markov Model on the training data. Almost 5 thousand such combination of states and sequences were obtained. Using Maximum likelihood Estimate and Baum Welch Algorithm appropriate Transition Matrix and Emission Matrix were calculated.

Transition matrix in the given system tells us about the probability of transition from one state to another i.e. if currently the system is in the open state then what is the probability that next state will be stopped state or Heavy traffic .Hence in the transition matrix shown below, we can see that probability of O->0 is 0.9197 and O->S.T is 0.08204 and since it's not possible to go to stopped (S)state from open(o) state so its probability is coming out to be 0,similarly all other possible transitions can be seen and validated.

```
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.

>> TRANS

TRANS =

      0.91975     0.080247            0            0
      0.02549      0.91961     0.054902            0
            0     0.036096      0.93449     0.029412
            0    0.0026042     0.054688      0.94271

>>
```

Figure 6.6: Transition matrix when studied with dense data set.

6.5 Testing of results using Synchro studio8 software:

Once classification of different road links was done, prediction of the future state was the next thing that should have been done. Sequence of the current traffic states were simulated on SYNCHRO STUDIO 8 software and the states of the main road were observed. Here a road network is shown in figure, in which traffic of 1st street and 2nd street meet traffic of Main Street so traffic state of Main Street is dependent on traffic states of 1st and 2nd street. Basically with this scenario we are trying to verify the results that we have taken in consideration in the HMM, there we have taken 64 combination but here there are 3 roads including the main road hence only 8 combinations are viable and hence they are being verified from this simulation network. For e.g. if we have open traffic on the 1st street and mild traffic on the second street according to that their os a possibility that main road will have either open or mild congestion state and hence this was verified from this software.

More to it if we have to do this same thing in the real scenario it can be done but it will be a little tedious task and it is not sure that we will be getting all the cases in the one go so here is the point where simulation software have an advantage and hence they are widely used for traffic networks.

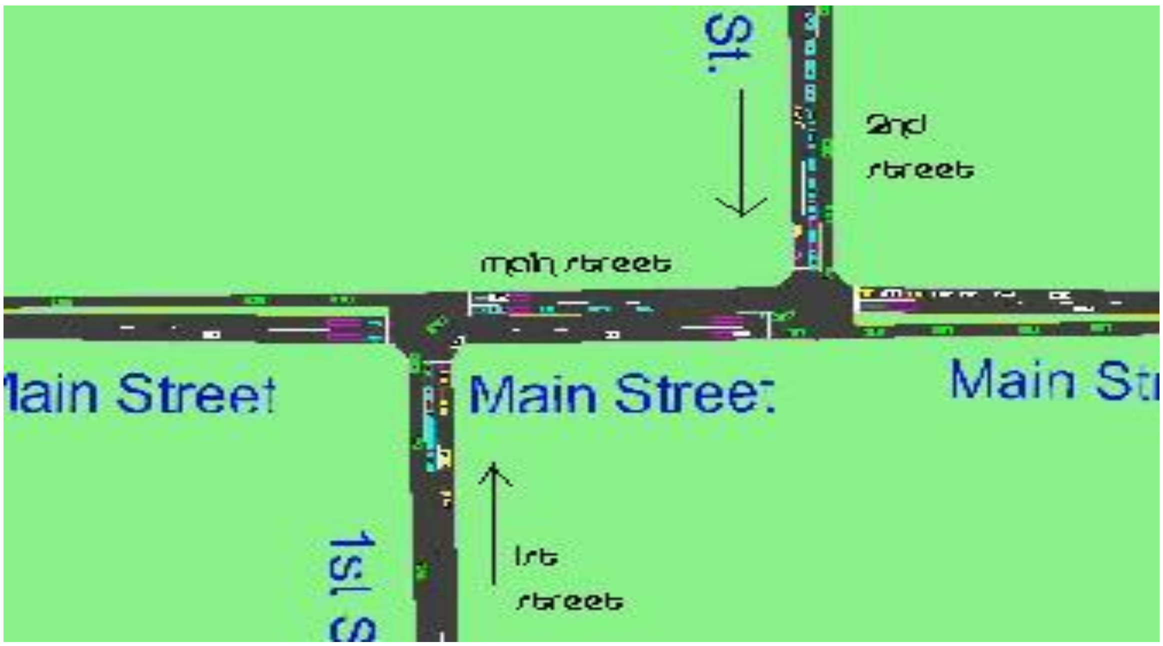


Figure 6.7: Road links

In the next figure red circles show total number of vehicles and blue circles shows moving vehicles so in the figure below, two vehicles are in stopped condition (absence of blue circles) and others are moving. According to our observation of different traffic conditions we can say traffic state of 1st street is mild.

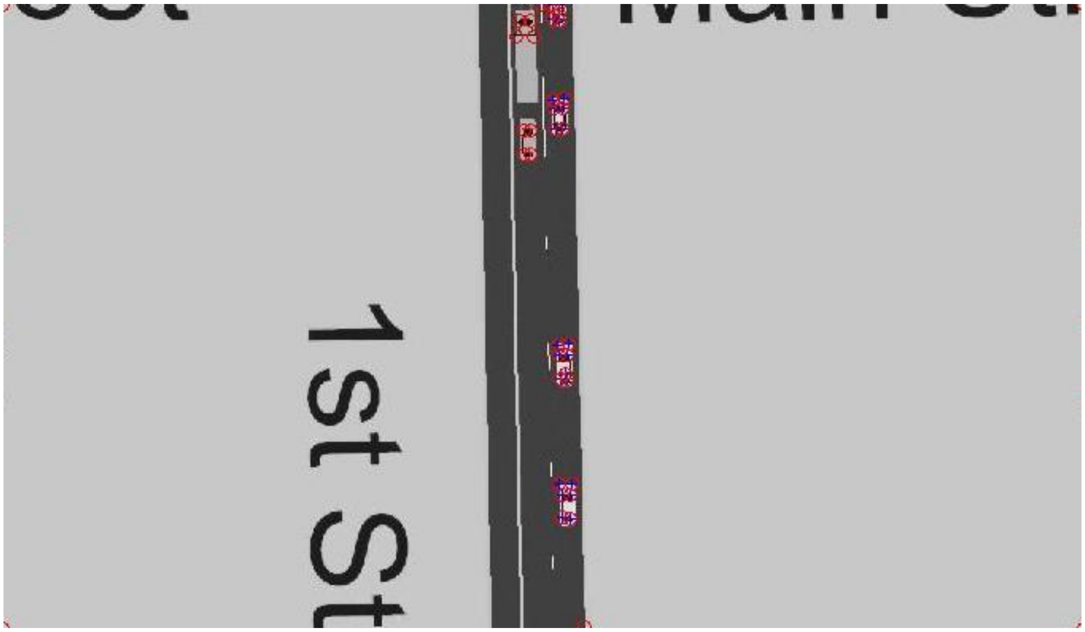


Figure 6.8: 1^{st} street

Similarly, in this figure traffic state of 2^{nd} street is shown. Red circles are much more than blue circles so it is in heavy state

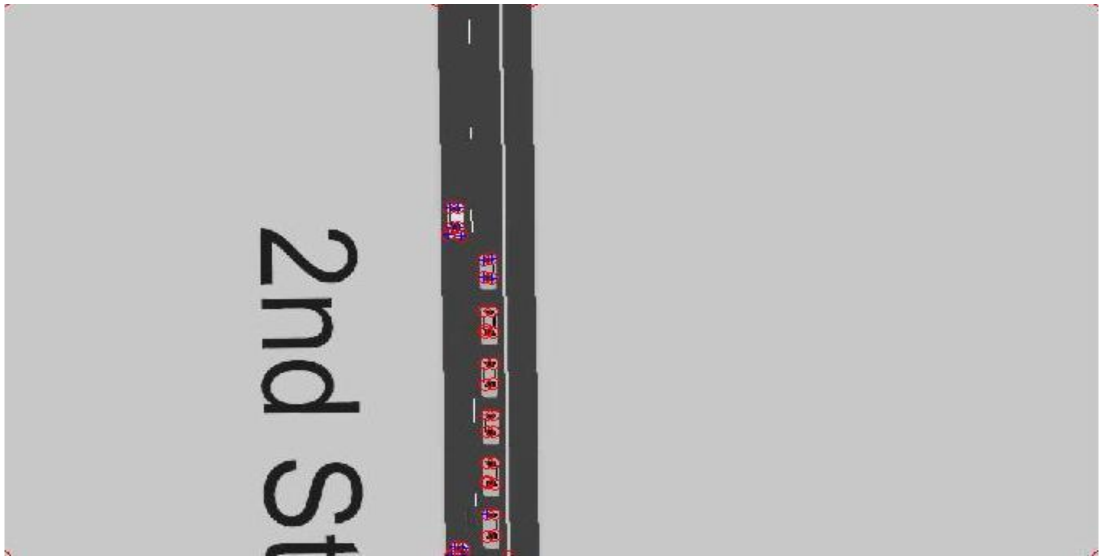


Figure 6.9: 2nd Street

From our observation we could conclude that there is a definite dependence of the traffic condition of the auxiliary or connecting road links on the main road that is in consideration. We can see in the image below that the traffic condition is mild in nature and is tending to be heavy traffic any moment.

Fig 6.10: Main Street

6.5.1 Validating HMM results using Synchro Studio 8:

In order to validate the results of the HMM transition matrix, this graph is simulated. In this graph on x- axis we have the no. of trials which we have made on the Synchro studio 8 simulator. And on y-axis there are 2 values i.e. 0 or 1, which signifies the correctness of the system. Now, In order to verify the results, we compare the learning phase results with the simulated environment results. Whenever the two results match, it shows that the system which we have developed is working efficiently and remains in state '1' else it deviates to state'0'.

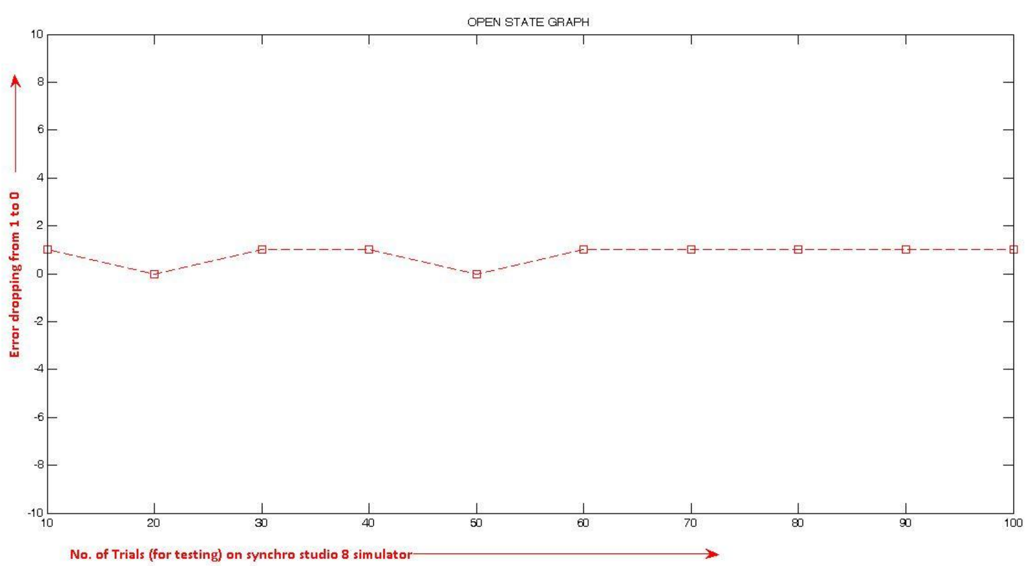


Figure 6.11: Error Graph for Open State.

The procedure is repeated for the open and slight congestion state as in the simulator we could get only these two conditions aptly, and in the similar way observations are compared and error graph is plotted.

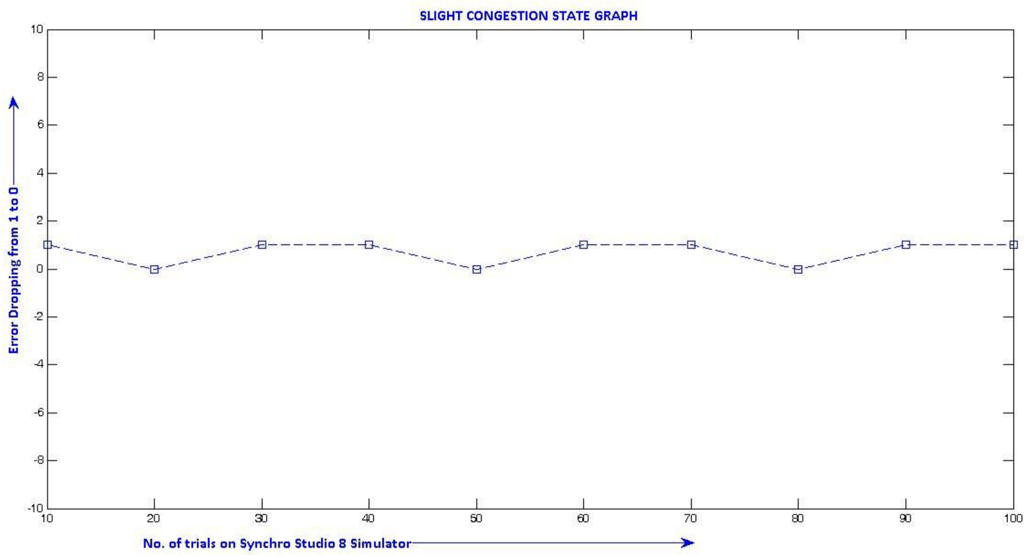


Figure 6.12: Error Graph for Slight Congestion State

Table 4: Efficiency of the algorithm for Open and slight congestion states.

	Open(100 trials)	Slight congestion(100 trials)
Efficiency	95.4	94.7

In Table 4 given above, columns describe the testing environment and rows tell us about the learning environment or the actual environment. Information about the number of times our system works in synchronization with the real environment has been depicted in the table.

Table 4 shows that 100 trials were made on the Synchro studio 8 simulator on the 8 minutes long video. After an interval of 5 seconds each, frames were tested with the learning data. With this, approx 95% efficiency has been achieved in case of open state and rest it has been wrongly predicted as slight congestion and hence giving us the error and similarly slight congested state has been predicted correctly approx. 94 times out of 100 and rest of the times it has been predicted as other states and hence giving us errors.

6.6 Future Scope:

This work has an important application in the real time traffic monitoring systems for predicting the traffic conditions of the roads given the current traffic conditions of the connecting roads. We can use IP camera network to get the traffic conditions of the connected roads.

6.6.1 Proposed Steps:

1: Installation of IP camera on the entire road links.

2: Capturing and simultaneous processing of the video.

3: In the processing part we will 1) Calculate the SIP and STIP points.

2) Classification of video frame into any of the four states.

4: Communication between cameras with the help of GPRS.

5: Prediction of the future state with the help of HMM.

6: Predicted state will be sent to the repository with the help of OGC standards from where any user could be able to retrieve it.

After the implementation of the IP camera module, next step is to send all the traffic information to the repository. For this there has to be a some common standard that is very well understood everywhere, so it is at this very moment that OGC(Open Geospatial Consortium) comes into play. The communication between camera network and repository is

done with the help of OGC standard. Standardization of OGC is basically in the development stage and many researchers are working towards it. In an Open Geospatial Consortium, Inc. (OGC)2 initiative called Sensor Web Enablement (SWE), members of the OGC are building a unique and revolutionary framework of open standards for exploiting Web-connected sensors and sensor systems of all types: flood gauges, air pollution monitors, stress gauges on bridges, mobile heart monitors, Webcams, satellite-borne earth imaging devices and countless other sensors and sensor systems.SWE presents many opportunities for adding a real-time sensor dimension to the Internet and the Web. This has extraordinary significance for science, environmental monitoring, transportation management, public safety, facility security, disaster management, utilities Supervisory Control and Data Acquisition (SCADA) operations, industrial controls, facilities management and many other domains of activity. The OGC voluntary consensus standards setting process coupled with strong international industry and government support in domains that depend on sensors will result in SWE specifications that will quickly become established in all application areas where such standards are of use.

Refrences

[1] J. A. Bilmes. A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models.TR-97-021, International Computer Science InstituteBerkeley CA, 94704, April 1998.

[2] R. Boel and L. Mihaylova. A particle filter for freeway traffic estimation. IEEE Symposium, June 2004.

[3] B.-J. Chang1, B.-J. Huang, and Y.-H. Liang. Wireless sensor network-based adaptive vehicle navigation in multihop-relay wimax networks. 22nd International Conference on Advanced Information Networking and Applications, pages 56{63, June 2008.

[4] P. Cheng, Z. Qiu, W. Qiu, and B. Ran. A particle filter based model for traffic state estimation using simple wireless network data. IEEE International

[5] R. Cucchiara, M. Piccardi, and P. Mello. Image analysis and rule-based reasoning for a trafficmonitoring system. IEEE Transaction on Intelligent Transportation System, 1(2):119{129, June 2000.

[6] R. Duda, P. Hart, and D. Stork. PatternClassification. Wiley Interscience, 2000.

[7] Y.-K. Jung, K.-W. Lee, and Y.-S. Ho. Content-based event retrieval using semantic scene interpretation for automated traffic surveillance. IEEE Transaction on Intelligent Transportation System, 2(3):151{163, September 2001.

[8] W. Kienzle1, B. Schfilolkopf1, F. A. Wichmann, and M. O. Franz1. How to find interesting locations in video a spatiotemporal interest point detector learned from human eye movements. 22nd International Conference on Advanced Information Networking and Applications, pages 56{63, June 2005.

[9] X. Li and F. M. Porikli. A hidden markov model framework for traffic event detection using video features. IEEE International Conference on Image Processing, 2004.

[10] B. Maurin, O. Masoud, and N. Papanikolopoulos. Monitoring crowded traffic scenes. IEEE Int. Conference on Intelligent TransportationSystems, pages 19{24, June 2002.

[11] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. International Journal of Computer Vision, 60(1):63{86, 2004.

[12] Y. Qi. Probabilistic models for short term traffic condition prediction. Ph.D. Thesis, M.Ap.Stat., Louisiana State University, May 2010.

[13] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In Proceedings of the IEEE, 77:257{286, 1989.

[14] D. Terzopoulos and C. Brown. Model-based vision for traffic scenes using the ground-plane constraint.Real-time Computer Vision Cambridge University Press, 1994.

[15] M. Tubaishat, Y. Shang, and H. Shi. Adaptive traffic light control with wireless sensor networks. 1-4244-0667-6/07/ IEEE, pages 187{191, June 2007.

[16] M.A.T. Figueiredo and A.K. Jain. Unsupervised learning of finite mixture models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(3):381–396, Mar 2002.

[17] G. Willems, T. Tuytelaars, and L. V. Gool1. An efficient dense and scale invariant spatio temporal interest point detector. 22nd International Conference on Advanced Information Networking and Applications, pages 56{63, June 2008.

[18] Y. Zou, G. Shi, H. Shi, and H. Zhao. Traffic incident classification at intersections based on image sequences by hmm/svm classifiers. IEEE International Conference on Hybrid Intelligent Systems, August2009.

[19] Laptev, I.: On space-time interest points. IJCV 64(2) (2005) 107–123

[20] S Indu, Sankalp Arora Santanu Chaudhury Asok Bhattacharyya" Road Traffic Model using distributed Camera Network " ICVGIP,2010

[21] J. J. Verbeek, N. Vlassis, and B. Kröse. Efficient greedy learning of Gaussian mixture models. *Neural Computation*, 5(2):469–485, Feb 2003.

[22] M. Hamouz, J. Kittler, J.-K. Kamarainen, P. Paalanen, and H. Kälviäinen. Affineinvariant face detection and localization using GMM-based feature detector and enhanced appearance model. In Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition, pages 67–72, 2004.

[23] Laptev, I. and Lindeberg, T. 2003a. Interest Point Detection and Scale Selection in Space Time. In Scale-Space'03, L. Griffin andM. Lillholm (Eds.), Vol. 2695 of Lecture Notes in Computer Science, Springer Verlag, Berlin, pp. 372–387.

[24] Laptev, I. and Lindeberg , T. 203b. Interest points in space-time. In Proc. Ninth International Conference on Computer Vision, Nice, France.

[25]A Report on" BAYESIAN CLASSIFICATION USING GAUSSIAN MIXTURE MODEL AND EM ESTIMATION:IMPLEMENTATIONS AND COMPARISONS" Information Technology Project Lappeenranta University of Technology Pekka Paalanen