



Intelligent Vehicular Traffic Light Control using Hidden Markov Model

Dominic Asamoah
Department of Computer
Science, KNUST, Ghana

Samuel Winful
Department of Computer
Science, KNUST, Ghana

Stephen Opoku Oppong
Department of Computer
Science, KNUST, Ghana

ABSTRACT

Traffic management continues to remain a major problem in big cities. Allocating limited resources, i.e. roads, to an increasing number of users with individual needs and objectives, turns out to be a highly complex in most cases.

This research uses Hidden Markov Model (HMM) as a component with unsupervised clustering scheme to determine the traffic situation of a road in a traffic video. An unsupervised clustering algorithm called Autoclass is applied to obtain the traffic density state (free, normal and congested) on motion features which are extracted from each frame. The three HMM models are constructed for each traffic state with each cluster corresponding to a state in the HMM. The result shows that this method can handle varying illumination and classify traffic density in a (Region of Interest) ROI of a traffic video.

General Terms

Algorithms, Artificial Intelligence, Image Processing

Keywords

Traffic management, unsupervised clustering, Hidden Markov Model, Autoclass, traffic density

1. INTRODUCTION

Urban Transportation have a number of challenges which includes high population densities in places, economic activities being concentrated more in urban areas and complex spatial structures that are not supported by transport systems. This complexity gets larger as the city grows thus implies a great danger if complexity is not efficiently and effectively controlled. Therefore, transportation challenges arise in urban areas if certain requirements are not met to satisfy urban mobility.

The late President of Ghana, West Africa, His Excellency Prof. John Evans Atta Mills, expressed his dismay about the traffic situation in Accra, the capital city of Ghana describing the traffic congestion on the roads as a “major headache” while inspecting a highway being construction near the Kotoka International Airport in Accra [1].

Vehicular congestion, motorization and the diffusion of the automobile are linked together and as well as the increase and demand for transport infrastructures. But the increase in mobility has not been matched with the supply of infrastructure to meet the demands of the growing condition

In order to solve the above-mentioned problem, many research groups have undertaken investigations to understand the problem and ‘the importance of development of new technological systems in transport’ [2], [3]. Traffic optimization is made up of different problems, from which

one of the most relevant ones is the traffic light cycles optimization.

This study aims to propose an intelligent traffic light system and introduces an additional component in the form of a Hidden Markov Model that statistically is used to monitor the traffic environment and to make timely decision in order to improve traffic light configuration. The system is based on defining three traffic cases (i.e. empty, normal and congested) that are used to provide the correct control signals required to make swift changes of the traffic system. The purpose of the HMM component is to predict the traffic situation several moments before it actually occurs.

2. LITERATURE REVIEW

This section provides literature review related to traffic light systems, highlighting some of the traffic light models that were developed to improve traffic light efficiency.

2.1 Artificial Neural Network

In the past years, Artificial Neural Networks (ANN) [4], [5] has been an object of great attention as an alternative approach to vast range of problems. Developments occurred at the algorithms and methods levels and on their application to different problems. Despite the continuing research works and expected further developments, they can be considered as an established technique with recognized advantages and limitations.

The advantages of using various ANN types are based on the multiple and possibly parallel processing elements, natural and easily inclusion of nonlinear functions, the use of multiple adjustment parameters distributed over a network. These characteristics are combined with a methodology for finding the values of the network parameters that can be based on existing data, example or instances of the problem rather than on an explicit identification of the variables and interrelations of these variables required in a mathematical formulation of the problem. These characteristics result in clear advantages when considering an approach to model a vast range of complex nonlinear problems in different areas. This research also covers the problem of traffic situation that can be considered as non-linear situation since no traffic situation can be easily predicted but by introducing HMM, it is hoped that the non-linear aspect of the traffic can be resolved.

2.2 Wireless Communication Based Traffic Light Models

Recently, Wireless Sensor Network (WSN) [6], [7] has made a breakthrough and has become a great force in modeling of traffic. This methodology employs the use of a sensor in each vehicle, distributed servers on each crossing, and embedded tracking sensor on the road which makes it very difficult and



expensive to run. Ben-Jye and William [7], [8] in their work gathered data from moving vehicle, analysed the collected and explained how a traffic navigation model works through WSN. Under this model, distance between the vehicles, average speed of the road and cost of each road link is considered and calculated. The drawback of this system is that the WSN should be placed in each vehicle. Malik, and Zou et al [9], [10] developed a traffic light control system which mainly centered on pedestrians crossing and reducing waiting time using embedded magnetic loop sensors on each lane of the road.

2.3 Kalman Filter Methods

Yang et al [11] in their paper on short term traffic prediction proposed a Recursive Least Square (RLS) approach to be used which can adapt to a possible changed pattern quickly. This approach used a 'forgetting factor' which assigns higher weights to new observations and exponentially forgets the old observations. Further, the recursive approach was explored further to a more general state space model by using the Kalman Filter (KF). The KF was based on the Bayesian rule and the maximum likelihood method. The proposed model was evaluated using real-time five - minute aggregated loop detector data collected from I-405, California. At the end of the study, the results showed that after adaptive learning of the training data in the 95% confidence interval of less than 10 mph, most of the true speed value fell in the validation process.

Xia and Chen [12] using the Kalman Filter method developed a dynamic short-term corridor travel time prediction model. This method combines a seasonal autoregressive integrated moving average model with a multi-step-ahead prediction of traffic condition. An embedded adaptive Kalman filter and an on-line corridor travel time prediction model is designed to adjust the prediction error based on traffic flow data that becomes available in real time. Test results show that this method is able to capture the traffic dynamics and provide more accurate travel time prediction.

2.4 Time Series Methods

Ishak and Al-Deek, [13] developed a generalized linear model to evaluate, the performance of the non-linear time series traffic prediction system in a wide experiment conducted in Orlando, Florida on the 40-mile corridor of I-4. This experiment was conducted under various model parameter and traffic conditions. This linear model was used to test the effects of the prediction horizon, prevailing level of congestion, their interaction on the model relative error of traffic speed prediction and the rolling horizon. The study concluded that the performance of the model declines very quickly as congestion increases and all the parameters of the tested model have significant influence on the relative error. Shorter prediction and rolling horizon are more favorable during congestion which was indicated by the significant interaction between model parameter with congestion index. It was also discovered that the model has a slight tendency to underestimate the travel times after the performance of the system was evaluated in terms of relative error of predicted travel time using the predicted speed information

Recently, Kaveh [14] evaluated the performance the simple moving average method with constant weights and two adaptive moving average methods in terms of how it predicts average travel speeds up to 10 minutes ahead of time. The method using optimized weights produced slightly better

predictions at a higher computational cost at the end of the study. It was also found out that the quality of predictions and the time for which predictions are made is inversely related. They tend to overestimate travel speeds in slow-down and underestimate them when the congestion is clearing up and speeds are increasing.

2.5 Other Models Studied

Vlahogianni [15] in his paper sought to bring out the importance of the knowledge of the statistical behavior of traffic dynamic associated with traffic conditions as well as the boundary traffic conditions associated with the changes in traffic patterns using discontinuous dynamic behavior of traffic data. To make the traffic prediction procedure simplified, a multilayer strategy to identify and further cluster the traffic patterns using 13 days of 90 seconds volume was proposed. Collected occupancy data in a highly congested urban area in the metropolitan area of Athens was used for the study.

Tan, [16] in an attempt to predict short term traffic flow based on the exponential smoothing (ES), moving average (MA), and neural network (NN) and autoregressive MA (ARIMA) models developed an aggregation approach. The predictions taken from the various models are used as the basis for the Neural Network model in the aggregation stage, and the final prediction is the output of the trained NN.

2.6 Hidden Markov Model

The methods used in the Hidden Markov Model approach is discussed which includes certain parameters including variables selection for traffic states definitions, construction of HMMs and how they can be used to determine traffic conditions.

2.6.1 Discrete Markov Process

The discrete Markov process produces conditional or transition probabilities of a finite number of future events without requiring any historical knowledge of the data, only the present state. The one-step transition probability is a probability that a future state will occur immediately following the present state. The one-step transition probability gives the conditional probability of visiting a particular state in the next transition given the current state. Firstly, a set of traffic states needs to be defined. Let $S(t)$ denote the state of the system at time t . The state space is the set of all the possible system states. The traffic states are numbered by integers from 0 to n . The state space is therefore

$$X = \{0, 1, 2, \dots, n\}.$$

Let $P_i(t) = Pr(S(t) = i)$ be the probability that the system is in state i at time t . The state probability distribution is denoted by

$$P(t) = (P_0(t), P_1(t), \dots, P_n(t)).$$

The process starts in one of these states and moves successively from one state to another. Each move is called a step. If the traffic state is currently in state S_i , then it moves to state S_j at the next step with a probability denoted by P_{ij} ; this probability does not depend upon which states the traffic was in before the current state. The probabilities P_{ij} are called transition probabilities. The process can remain in the state it is in, and this also occurs with probability P_{ii} . The transition probability matrix generated from empirical data can be used to estimate the expected density and number of vehicles.

The transition probabilities of the Markov process

$P_{ij} = Pr(S(t + \tau) = j | S(t) = i)$
 $= Pr(S(t + \tau) = j | S(\tau) = i, S(u) = s(u), 0 \leq \tau < t)$
 can be arranged as a matrix

$$P(t) = \begin{bmatrix} P_{00}(t) & P_{01}(t) & \dots & P_{0n}(t) \\ P_{10}(t) & P_{11}(t) & \dots & P_{1n}(t) \\ \cdot & \cdot & \dots & \cdot \\ P_{n0}(t) & P_{n1}(t) & \dots & P_{nn}(t) \end{bmatrix}$$

The square matrix $P(t) = (P_{ij})$, $i, j \in X$ is called the one-step transition matrix, and since when traffic is in state i at time 0, it must be either in state i at time t or to make a transition to a different state $j \in X$.

The following condition must be satisfied

$$\sum_{j=0}^r P_{ij} = 1$$

The size of the transition matrix is $n \times n$, where n is the number of defined states.

A general assumption for vehicular traffic flow is that states too far back in the past will not give any more information than the current and immediate states. If a series has the attribute that, given the current and $n - 1$ preceding states, the future state is independent of the states prior to the present state, the series can be called a Markov process with n^{th} order. In this paper, a Markov process is employed to predict the traffic states when the traffic light is changing to green.

2.6.2 Markov Chains

Many real-world problems involve the study of systems that vary in time at a random manner. Mathematical models of such systems are known as stochastic processes. A stochastic process can be described as any collection of random variables S_t , $t \in T$, defined on a common probability space. If the time parameter T is an interval, the process is called a continuous parameter process, while it is a discrete parameter process if T is a subset of integers. The values of the variables S_t form a set is known as state space of the process.

Stochastic processes based on their properties can be classified into various groups. Those possess the property that given the present state of the process, the past history does not affect conditional probabilities of events defined in the future are called Markov Process. Markov Chains are discrete parameter Markov Processes whose state space is finite or countable infinite.

Consider a system that can be in any one of finite or countable infinite number of states (s_1, s_2, \dots, s_n) . Let S represents this set of states, which is the state space of the system. The system is observed at the discrete moments of time thus $t = 1, 2, 3, 4 \dots T$ and S_t represents the state of the system at the time t . For a first order Markov Chain, the probabilistic description of the system is truncated to just the current and the predecessor. In other words the probability of the state of the system S_t is s_t at the time t only depends on the predecessor state at time $t-1$, not all the past states, as described below:

$$P(S_t = s_t | S_{t-1} = s_{t-1}, S_{t-2} = s_{t-2}, \dots, S_1 = s_m) = P(S_t = s_t | S_{t-1} = s_{t-1}) \quad (1)$$

The right hand side of equation (1) is called the transition probability (or coefficient) from states S_i to state S_j , represented by a_{ij} . In the study, only those processes in which the state transition probabilities are independent of time are

considered, thereby leading to the set of stationary state transition probabilities a_{ij} of the form

$$a_{ij} = P(S_t = s_j | S_{t-1} = s_i), \quad 1 \leq i, j \leq N \quad (2)$$

Since the state transition coefficients obey standard stochastic constraints, they have the following properties

$$a_{ij} \geq 0 \quad (3)$$

$$\sum_{j=1}^N a_{ij} = 1 \quad (4)$$

2.6.3 Hidden Markov Model

The Markov Chains can also be called an observable Markov Model since the output of the process is the set of states at each instant of time, where each state corresponds to an observable event. The state of the process is known at each instant of time. Extending the concept of Markov Chains to the cases where the observation is a probabilistic function of the state will result in a Hidden Markov Model. For a Hidden Markov Model (HMM), the state of the system at a discrete moment of time is unknown, only a physical event is observed, where the observation is a probabilistic function of the state. An HMM is a doubly embedded stochastic process with an underlying stochastic process that is not observable (it is hidden), but can only be observed through another set of stochastic processes that produces the sequence of observations [17].

An HMM is characterized by the following five elements:

- i. N , the number of states in the model. The individual states are denoted as $s = \{s_1, s_2, s_3, \dots, s_N\}$, and the state of the system at time t as S_t
- ii. M , the number of distinct observation symbols. The individual symbols are denoted as $O = \{O_1, O_2, O_3, \dots, O_M\}$
- iii. The state transition probability matrix $A_{N \times N} = \{a_{ij}\}$ ($1 \leq i, j \leq N$),
- iv. The observation symbol probability matrix $B_{N \times M} = \{b_{jk}\}$, where

$$b_{jk} = P\{O_k \text{ at } t | S_t = s_j\}, \quad 1 \leq j \leq N, 1 \leq k \leq M, t = 1, 2, 3, \dots \quad (7)$$
- v. The initial state distribution vector $\pi = \{\pi_i\}$, where

$$\pi_i = P\{S_1 = s_i\}, \quad 1 \leq i \leq N \quad (5)$$

It should be noted that the state transition probability matrix A and observation symbol probability B are independent of time. The five elements are defined by HMM as stationary model. In most real-world applications of HMMs, the model parameters N and M are known as beforehand, while the three probabilistic measures A , B , and π could be known or unknown. For the purpose of convenience, γ is used to indicate the probabilistic parameter set of the model.

$$\gamma = \{A, B, \pi\} \quad (6)$$

2.7 Model Predictive Control

Model Predictive Control is a model-based optimization control method for urban traffic networks. The MPC method applies Optimal Control in a rolling horizon way. At each step only the first part of the optimal control solution is used. Then the horizon is shifted one step forward and then the optimization is started again with new information of the measurements. The optimization is based on the prediction

model of the process and an estimate of the future traffic disturbance.

The advantages of MPC are that it can deal with the uncertainty of the process, which can be caused by the unpredictable disturbance, changing parameters over time and model mismatches in the prediction model. Another advantage is that you can easily change the prediction model.

The MPC framework has three typical features: The prediction model, optimization of the control input, and the rolling time horizon. These three steps can be described as follows

3. METHODOLOGY

Phase 1: Initially image acquisition is done with the help of video camera. First image of the road is captured, when there is no traffic on the road. This empty road's image is saved as reference image at a particular location specified in the program. RGB to gray conversion is done on the reference image. Now gamma correction is done on the reference gray image to achieve image enhancement. Edge detection of this reference image is done thereafter with the help of horizontal and vertical filters and normalization.

Phase 2: Images of the road are captured. RGB to gray conversion is done on the sequence of captured images. Now gamma correction is done on each of the captured gray image to achieve image enhancement. Edge detection of these real time images of the road is now done with the help of horizontal and vertical filters and normalization.

Phase 3: After edge detection procedure both reference and real time images are matched and traffic lights can be controlled based on percentage of matching (Effective density). If the matching is between 0 to 10% - green light is on for 90 seconds. If the matching is between 10 to 50% - green light is on for 60 seconds. If the matching is between 50 to 70% - green light is on for 30 seconds. If the matching is between 70 to 90% - green light is on for 20 seconds. If the matching is between 90 to 100% red light is on for 60 seconds.

3.1 Architecture Design

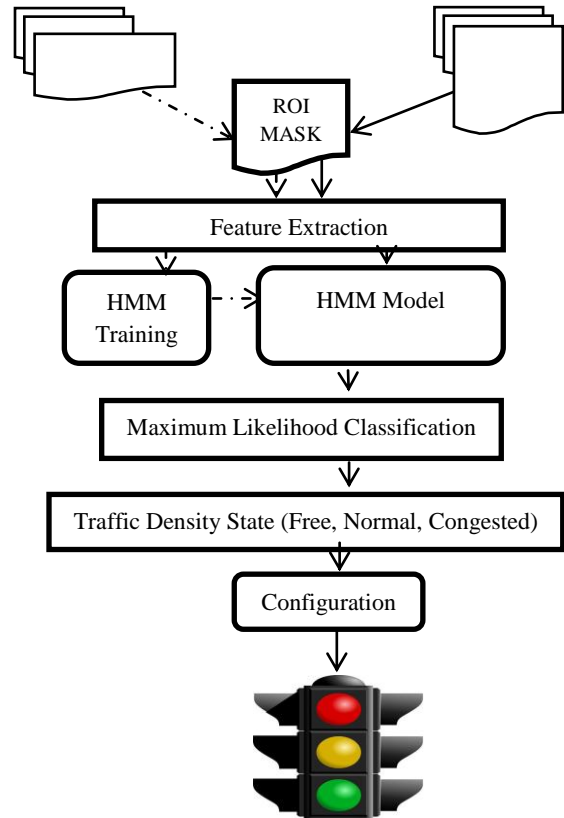


Fig 1: System Design

3.2 Intersection Structure

For the model used, figure shows the structure for an intersection. On each incoming and network link right or can go straight. Each intersection with four incoming links has in total twelve different traffic lights. The twelve different traffic lights are divided into four different groups. The lights in one group are served in the same way, so the lights in one group are non-conflicting flows. When one light in a group is green all lights in that group are green. When the lights in one group are green the lights in the other groups are red. The four different groups are shown in figure 2.

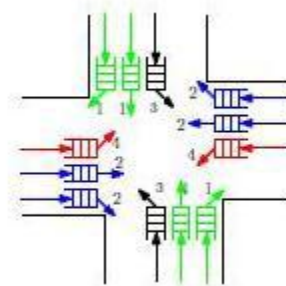


Fig 2: Serve 12 flows in 4 combinations.

3.3 HMM Framework

A Hidden Markov Model (HMM) can be viewed as a stochastic finite state machine where each state transition is determined probabilistically. A HMM is represented by: the number of states in the model, an observational symbol probability distribution (e.g. Gaussian) of each state, and a

state transition probability distribution with an initial state distribution.

In the HMM framework, the feature vectors extracted from the ROI area of video sequences and are grouped into three groups according to the three pre-defined labels in the training video sequences. These three groups of feature vectors are used to train three HMMs with each of them corresponding to one of the three predefined traffic density states. In the training stage, for each group, all the collected feature vectors are clustered by an unsupervised clustering algorithm called AutoClass proposed by Cheeseman and Stutz. The use of AutoClass this algorithm (Autoclustering) is able to reflect the impact of illumination changes at the feature vectors. For each resulting cluster of each group, a multivariate single Gaussian distribution is used to represent the observational probability of each state in one HMM. Then, each feature vector belonging to the same cluster group is given a unique cluster *ID*, by which the sequences of the cluster *ID* are used to obtain the initial state distribution and the transition probabilities of each HMM. Finally, the initial state distribution, transition probabilities, and observational probabilities are used to form the parameter sets for the three HMMs respectively.

Constructing the HMMs based on the clustering information means that the structure of each HMM will be “discovered” by the clustering algorithm and might not necessarily be the same as widely used HMM structures like the ergodic model and the left-right model.

3.4 Unsupervised Clustering

An unsupervised clustering scheme is used here because, in practice, it is difficult to tune the number of clusters in each HMM to reflect the impacts of the illumination changes on the low-level feature vectors, given the different camera setup and road surface conditions. Many clustering techniques such as K-Means and C-Mean [18] can also address such feature groupings. However, their performance depends on the initial parameters as well as the number of clusters *K*.

The following steps describe how the AutoClass is implemented in our paper,

- i. After feature extraction, a feature vector $X_n = \{X_{0n}, X_{1n}, X_{2n}, X_{3n}\}$ is extracted from each video frame *n*, where *D* is the number of dimensions of the feature vector (21 here).
- ii. Separate and label a video shot manually so that all frames in the video shot belong to a same traffic density state {Empty, Normal, Congested}
- iii. According to the labels, frames from all video shots are grouped into three groups with each group corresponding to a traffic density state. As a result, each video shot in each state group contains a mini-sequence that represents a single

empty/normal/congested traffic density state. The AutoClass is then applied on each of these groups, respectively.

- iv. For each group, given the parameter set $V_n = \{V_1, V_2, \dots, V_J\}$, for *J* clusters, the single multivariate Gaussian p.d.f with model parameters V_j for each cluster is:

$$P(X|X \in C_i, T, V_j, S) = \frac{1}{\sqrt{2\pi}^d \sqrt{\det(\Sigma)}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)} \quad (7)$$

Where,

μ - Mean vector of the p.d.f of cluster *i*,

Σ - Covariance matrix of the p.d.f of cluster *I*,

$V_j = \{\mu, \Sigma\}$ - Gaussian model parameters

- Type of the probability model applied (Gaussian here),

S - The whole space described by *T*.

- v. Set a threshold value $\theta = \delta$ such that a cluster C_θ is considered as a weak cluster if the number of data points that falls into C_θ is below δ percent. Subsequently, the data-points in C_θ are reassigned to other clusters. Clusters affected by this have their mean μ and covariance Σ updated accordingly.
- vi. Go back to step 4 until no weak classes remain and all four frame groups have been clustered.

4. IMPLEMENTATION AND RESULTS OBTAINED

4.1 HMM Framework

To determine the traffic density, the features which are low level in nature is taken from the captured traffic videos and is then modelled using the Hidden Markov Model (HMM).

In make processing easier, a ROI is indicated manually in the video frames across whole video sequence. The ROI is currently marked to be a lane of the road. The remaining processes which are feature extraction, HMM training, and traffic density state classification are done based on the ROI only.

Three traffic density states which are Empty, Normal and Congested are specified based on the percentage of the ROI surface being covered by vehicles. An example of the three states and illustrated as shown below.



Fig 3: Examples of the three traffic density states, Empty, Normal and Congested

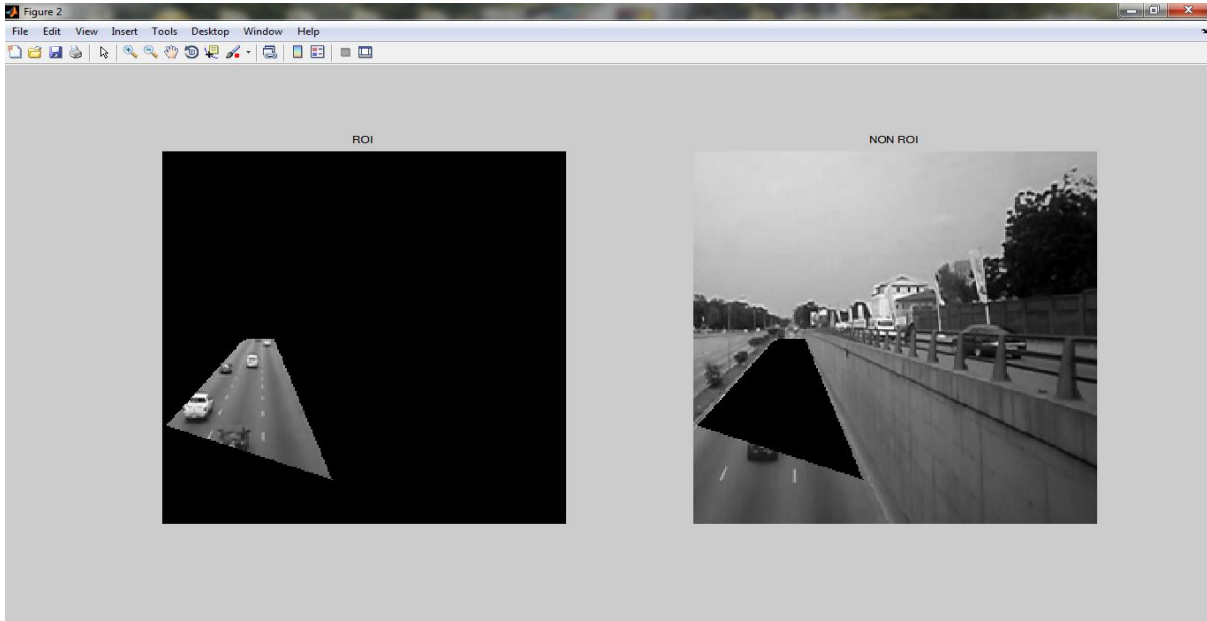


Fig 4: Example of a ROI area

Thus, our proposed algorithm can be easily applied on any other area that required monitoring by just shifting the ROI box to the area. One example of a ROI is shown in Figure 4, and non-region of interest.

4.2 Road Traffic State

As shown in Figure 2, three states are used to describe the traffic density in a ROI area of an intersection. According to the percentage of the ROI area occupied by the vehicle, three traffic density states, i.e. Empty/Normal/Congested states, are defined:

- Empty — less than 30% of ROI surface is occupied by vehicles,
- Normal — between 30% and 70% of ROI surface is occupied by vehicles,
- Congested — more than 70% of ROI surface is occupied by vehicles.

An example of the three states and the ROI is illustrated in Figure 3. Therefore, the objectives of this paper focus on the modeling of these three traffic density states under different traffic, and then recognizing the state of traffic density of each traffic video shot in real-time by model matching through a ML algorithm. In this paper, a traffic video shot is defined as a mini traffic sequence that is manually segmented out from a traffic video sequence. The system flow of the HMM is shown in Figure. 1, which comprises low-level feature extraction, unsupervised clustering, HMM training, and real-

time traffic density state determination three components. To ease the processing involved, a ROI is manually drawn out to indicate the region to be processed in the video frames across whole video sequence. Currently, the ROI is specified to be a lane of the road. All processes thereafter, including feature extraction, HMM training, and traffic density state classification are conducted based on the ROI only.

4.3 Detecting Traffic Density - Daytime

During the daytime, the underlying intuition is that when there is no traffic on the road, it appears gray in color irrespective of the natural day light. When the road is filled with traffic, the amount of visible gray (empty road) in the picture reduces because of the majority of vehicles attributing a varied level of non-gray color. To perform traffic density estimation, we first use a simple polygon to manually mark the road segment area for the image analysis. For a given traffic video on a road segment, this is a one-time operation that explicitly specifies the region of interest for the analysis. Then, we convert the picture into an 8-bit grayscale and analyze the pixels within the marked segment area. For each value (0-255), we plot a histogram for the number of pixels that have each of the 256 different gray-scale values. We have verified that the gray of roads lies within the 135-165 range. Intuitively and backed by the analysis, if an histogram is constructed for the varying levels of gray in the picture, depending on the level of congestion of the road, a histogram would be observed to have a smooth gray level area as compared to a high peak expected in an empty road. With increasing traffic, the peaks

at 135-165 begin to reduce and the drop-off at either side is more gradual. By examining these histograms, we can easily estimate the traffic density on the roads. Figure 5 shows snapshots from a traffic feed (along with the corresponding histogram) in Ring road central, Accra. The image on the left shows no congestion and we observe that the corresponding

histogram shows peaks in the road gray areas. Similarly, the figure on the right shows a congested road and the corresponding histogram. Observe that the histogram is more evenly spread out, and does not peak in the gray areas as much as the case with low congestion.

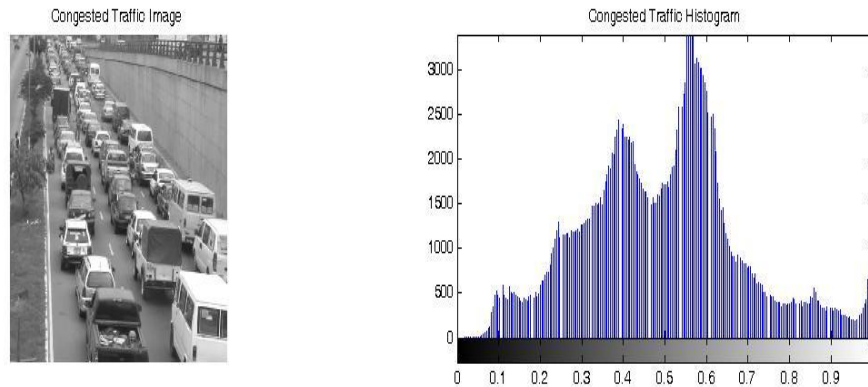


Fig 5: Congested Traffic Image and its corresponding Histogram

4.4 Detecting Traffic Density – Night Time

Detecting congestion at Night is a challenging problem due to some extreme factors. First of all, unavailability of light makes typical vehicle feature estimation techniques not applicable. Also, headlight counting becomes very difficult because of light reflection/refraction and alternate light sources such as billboards and traffic signal lamps. As explained in [19], a summary of vehicle identification technique used in the night time is discussed. The white pixel distribution in the image is estimated using the density estimation algorithm. The effects of reflected lightning from our thresholding process is reduced which significantly reduces the amount of false light present in the image due to the dissimilarities of intensities of both light source and the reflected/refracted light. Multiple images are thresholded to balance characteristics loss and also identify a suitable thresholding level, since successive thresholding levels affects the image noisiness in a progressive manner. A thresholding level of 0.8 was selected for the dataset of traffic image. After thresholding process, a gradient white light estimation process is done across the link length to calculate the traffic density.

The road segment is marked is typically along the middle of the image as used in this scenario as done in the day time estimation algorithm. Using the algorithm, the amount of white light in the image is calculated, reading a pixel line and registering the pixel measure towards the total white light in a weighted fashion. The weighted factor is required as the observer goes towards the far point of view. Light from multiple vehicles becomes difficult to distinguish and thus a same size segment contains a different number of vehicles as compared to the near point of view.

4.5 Traffic Light Configuration

Traffic light control can be done in different ways; you can use isolated control or coordinated control. Isolated control is used for the control of a single intersection and coordinated control is used in urban areas to control a network of intersections. When you use coordinated control, the

intersections in a specified area interact with each other. This is good for the traffic flow of the cars and reduces the overall waiting time, but the implementation of this is complicated.

If you use one of these control mechanisms an algorithm that determines the optimal state of a traffic light at each time step is needed. There are already some traffic control algorithms developed [20]. These can be divided into fixed-time algorithms and traffic response algorithms. Fixed time algorithms use historical data to make a fixed planning for the traffic lights for a specified future period. Traffic response algorithms interact with sensors that measure the number of cars that are waiting at the intersections, and it uses these real-time measures to control the traffic lights.

4.6 Performance Test

Computer with Intel i3 core processor with windows 7 platform was used in simulation.

Auto-HMM was tested on video sequences of a traffic intersection at ring road central in front of the Police headquarters Accra and traffic at Accra mall for day and night simulation respectively. Two set of video sequences were considered, a training set and testing set. 70% of the video sequence was used for training and the remaining (30%) for testing. The training set is labeled by assigning a traffic state to a whole video sequence. The training video shots covered day and night as shown in Figure 6. Video shots having same traffic state label are first collected and then clustered by AutoClass. This is to enable so each frame in a video shot to have an identical subcluster ID. After, all frames coming from video shots having the same label, they are used to train a HMM corresponding to the relevant traffic-density state. In the course of testing, each video shot is feed into the three HMMs individually. The HMM having the greatest probability is used to label the video shot. Also the threshold parameter δ of AutoClass is assumed to be 10 in the experiments. Also the length of each video shot processed by the ML estimator is set to 1 second with the feature extraction done on every 2 frames, i.e. the processing frame per second

(FPS) is 13, and 13 feature vectors are generated in each second given a 25 fps video sequence. Table 1 summarizes the video sequences used in evaluating the performance of the Auto-HMM algorithm.

To compare and determine whether the automated discovery process adapted used in our algorithm outperforms other setups with empirically chosen parameters, the same testing sequences is performed on two other algorithms i.e. the conventional HMM (C-HMM) which uses traditional Baum–Welch algorithm in training the HMM parameters without using any clustering scheme [17] and K-HMM the second algorithm which uses a HMM with a K-Mean clustering scheme. The parameters for number of states in the F-HMM and the Ks for K-Means in KHMM are set as three for full state frame group, three for high state frame group, three for low state frame group, and two for empty state frame group, which were obtained empirically through times of experiments to be able to generally optimize the testing results. The three algorithms were tested and compared for efficiency and reliability using video sequence with day and night conditions.

Table 1: Video sequences

Duration	Day	Night
No. of frames	5720	4850
ROI size	3 lane	3 lane

% EMPTY state	14.95%	38.35%
% Normal state	34.20%	47.90%
% Congested state	50.85%	15.95%
No. of frames for training	4004 (70%)	3395(70%)
No. of frames for testing	1716 (30%)	1455(30%)

The high training accuracy as shown in Table 1 proves that our algorithm converged adequately and also the significantly high accuracy during classification shows that our algorithm is not affected by Day and Night conditions which are different. Furthermore, the results also show that the automated state discovery could work reasonably well compared to the KMeans setup with empirically chosen parameters.

Table 2: Results of differing algorithms.

Period	Training Accuracy		Classification Accuracy	
	AutoClass %	KMeans %	AutoClass %	KMeans %
Day	95.4	96.9	98.9	98.8
Night	97.8	98.3	99.5	99.3

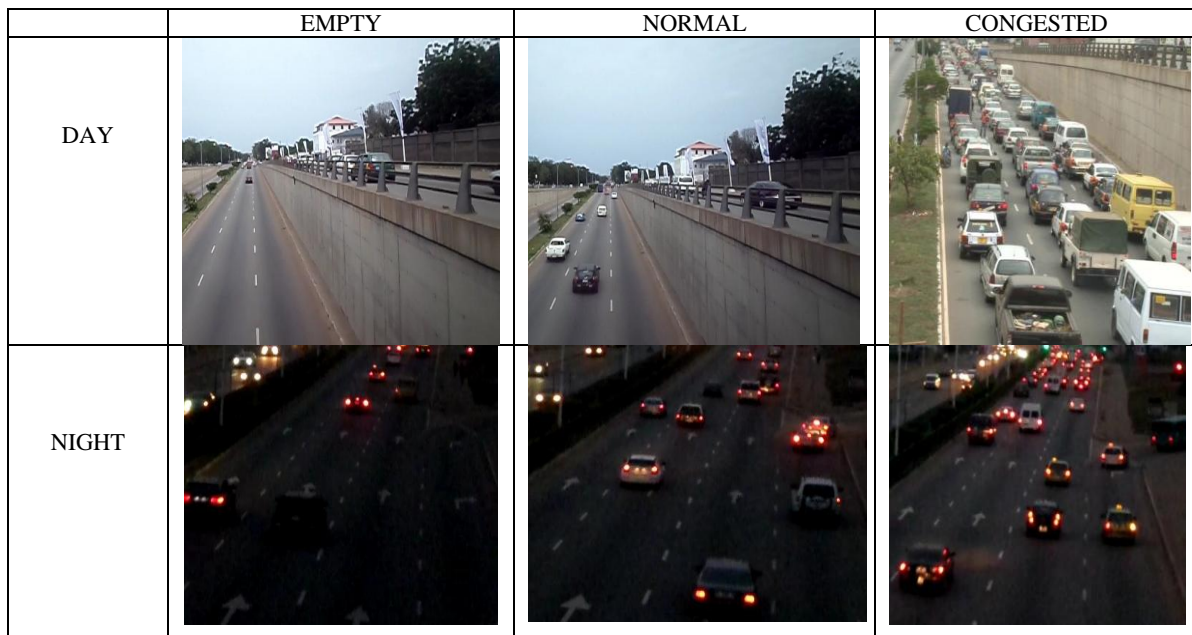


Fig 6: Samples of Training Data Set

5. CONCLUSION AND FUTURE WORK

Information technology (IT) now in the early stages of transportation system has transformed many other industries, from health care to education to government. While many others think improving a country’s transportation system solely means building new roads or repairing aging infrastructures, IT is the future of transportation lies not only in concrete and steel. This is so because IT helps elements within the transportation system vehicles, roads, traffic lights, message signs, etc. to become intelligent by embedding them

with microchips and sensors and enabling them to talk with each other through wireless technologies. This study based on the Hidden Markov Model, attempted to solve the hurdles of road traffic congestion by finding the traffic density through the design and implementation of an intelligent system; to monitor and control traffic lights. A way of determining the traffic density has been presented with reasonably good results under day and night condition. However, there is the need to find a way to automatically label the traffic sequence so as to determine the traffic flow density for the traffic light gives different time spanned for the green light in different traffic states.



6. REFERENCES

- [1] Hmm M., 2011. Ghana Web. Retrieved September 30, 2012, from Ghana Web Site: <http://www.ghanaweb.com/GhanaHomePage/NewsArchive/artikel.php?ID=221855>
- [2] Jarašūnienė A. 2004. “*The Importance of Development of new technological systems in Transport means*”, p. 233–236
- [3] Jarašūnienė A. 2007 “*Research into intelligent transport system (ITS) technologies and efficiency*”, Journal of Vilnius Gediminas technical university, Lithuanian Academy of Science., Vol. XXII, No 2 (ISSN 1648-4142), 61–69.
- [4] Patel M. and Ranganathan N. 1996. “*An Intelligent System Architecture for Urban Traffic Control Application*”
- [5] Patel M. and Ranganathan N. 1996. “*An Intelligent Decision-Making System for Urban Traffic-Control Application*”, IDUTC
- [6] Verma V. et al, 2008. “*Intelligent transport management system using sensor networks*”, IEEE Intelligent Vehicle Symposium Eindhoven University of Technology, pp. 4-6.
- [7] Williams G., Tuytelaars T., & Gool L., 2008. “*An efficient dense and scale invariant spatio temporal interest point detector*, International Conference on Advance Information Networking and Applications, pp. 56-63.
- [8] Chang B., 2008. “*Wireless Sensor Network-based Adaptive Vehicle Navigation in Multihop-Relay*”, WiMAX Networks. IEEE AINA.
- [9] Malik T., 2008. *Wireless Sensor-Based Traffic Light Control. Las Vegas, NV., IEEE Consumer Communications and Networking Conference*
- [10] Zou Y. et al, 2009. “*Traffic Incident classification at intersections based on image sequences by HMM/SVM classifiers*”, IEEE International Conference on hybrid Intelligent Systems
- [11] Yang F. et al, 2004. “*Online Recursive Algorithm For Short-Term Traffic Prediction*”. Transportation Research Record No 1879, Transportation Research Board, 1-8.
- [12] Xia J. and Chen M., 2009. “*Dynamic Freeway Corridor Travel Time Prediction Using Single Inductive Loopdetector Data*”. Transportation Research Board, Transportation Research Board 85th Annual Meeting, Washington D.C,
- [13] Ishak S. and Al-Deek H., 1856. “*Statistical Evaluation of I-4 Traffic Prediction System*”. Transportation Research Record, 16-24.
- [14] Kaveh F., 2010. *Evaluating Moving average Techniques in Short-Term Travel Time Using an AVI Data Set*, Transportation Research Board Annual Meeting, Washington D.C., 10-3144
- [15] Vlahogianni.E., 2006. K. M. Patterns in short-term Urban Traffic Flow: *A Framework for Statistical Detection and Identification*. Transportation Research Board 85th Annual Meetings, Washington D. C.
- [16] Tan M., 2009. “*An Aggregation Approach to Short-Term Traffic Flow Prediction*”. IEEE Transactions on Intelligent Transportation System, 10(1), 60-69.
- [17] Rabiner L., 1989. “*A tutorial on hidden Markov models and selected applications in speech recognition*”. Proc. IEEE 77, 257–286.
- [18] Sun, S., C.Z, 2006. “*A Bayesian network approach to traffic flow forecasting*”. IEEE Trans. Intell. Transp. Syst.7(1), pp. 124-132.
- [19] Jain V., 2012. *Traffic density estimation for noisy camera sources*. TRB 91st Annual Meeting, Washington D C.
- [20] Papageorgiou M., 2003. “*Review of road traffic control strategies*”. Proceedings of the IEEE, 91 (12), pp. 2043-2067.