The development of a smarter mode of transportation using AI to reduce travel and waiting times

Priyank Singhal, Karthikeyan M. P., Pramod Kumar Faujdar

Abstract Smarter mode of transportation (SMT) rely heavily on real-time traffic management. Safety on the roads is only one of many benefits brought forth by advancements in dynamic traffic management systems for congested metropolitan areas. In this study, we present the design and implementation of a Bayesian Belief Networks (BBN) traffic control system that is both flexible and reliable. The use of knowledge-based systems as a framework for making decisions in real-time has gained widespread acceptance. Since traditional dynamic controllers relied on sensors with their own set of drawbacks, we may employ vision sensors (such as cameras) to get around these problems. Computing based on images and videos is very useful for gauging traffic volumes. The present traffic management system at the road junction was found wanting, thus a new system was developed and put into place to alleviate the congestion. Lab VIEW and MATLAB are used to measure how well the suggested framework works. Extensive simulations utilizing the suggested method show that it reduces waiting time and speeds up movement on average compared to controllers employing traditional sensors.

Keywords: traffic density measurement, controller, bayesian belief networks, traffic lights, vision computing

1. Introduction

The creation of a more advanced form of transportation will include incorporating a wide range of technological breakthroughs and inventions to increase productivity, decrease congestion, and improve the experience of traveling as a whole (Zhang 2020). It is intriguing to think about the possibility of designing a more advanced form of transportation that uses artificial intelligence (AI) to shorten both trip and wait times. Artificial intelligence can completely transform transportation networks by streamlining routes, improving traffic flow, and increasing overall efficiency (Lilhore 2022). The movement toward environmentally friendly and electric cars is essential to intelligent transportation systems. Electric vehicles, such as automobiles, buses, and bicycles, have the potential to drastically cut carbon emissions and the reliance on fossil fuels, thus contributing to a cleaner and greener transportation system (Soori 2023). Cars that are connected and the transportation infrastructure may interact with one another as well as with the sophisticated sensors, communication technologies, and artificial intelligence that are equipped with these connected cars. Sharing data in real-time, working together while driving, and improving overall traffic management are all made possible due to this. There is a possibility that autonomous cars, which depend on artificial intelligence (AI) and sensors to function without any interaction from a human driver, would improve safety, efficiency, and the flow of traffic (Leung 2019). Using artificial intelligence (AI) and big data analytics, transportation agencies and service providers can collect and analyze massive volumes of data from a variety of sources. These data may determine trends, forecast travel demand, optimize routes, and make choices based on the collected data to increase operational efficiency and decrease travel times (Cao 2019). The development of intelligent transportation also emphasizes preserving the environment and being environmentally responsible. This includes the promotion of walking and cycling infrastructure, the encouragement of the use of public transit, and the integration of renewable energy sources into various modes of transportation (Paiva 2021). Artificial intelligence (AI) can potentially play a key role in lowering travel and waiting times through the optimization of transportation networks. AI systems may examine historical traffic data, current weather conditions, and information received in real-time from sensors to make predictions about traffic patterns and congestion hotspots. This information may be utilized to improve traffic flow, propose alternative routes, and maximize route optimization, which will ultimately result in a reduction in travel times (Offiaeli 2021). Algorithms that are driven by AI may take into account real-time traffic data, the state of the roads, and other variables to determine
which routes are the most time- and fuel-efficient for particular cars. AI has the potential to shorten travel times and steer clear of crowded places because of its ability to dynamically update routes depending on current circumstances (Abduljabbar 2021). AI may be used to improve the efficiency of transportation systems sensitive to changes in demand, such as ridesharing and micro transit. AI can reduce waiting times and offer more effective transportation alternatives since it can analyze passenger demand and dynamically change routes and vehicle allocations (Witlox 2022). Users can obtain real-time travel information from systems that AI drives. This information may include public transportation timetables, traffic updates, and anticipated trip times. AI can assist passengers in making well-informed judgments and selecting the modes of transportation or routes that will save them the most time and effort by providing them with information that is both accurate and up-to-date (Fridgen 2021). Artificial intelligence can monitor traffic patterns and locate places of congestion by analyzing data collected in real time from various sources, such as traffic cameras, GPS devices, and sensors. These data may be used to vary the timing of traffic lights timings dynamically, redirecting cars, and recommend other routes to reduce travel times (Liu 2023). AI-powered routing algorithms can take into account a variety of criteria, including current traffic conditions, road closures, and user preferences, to choose the path that will be the most time and energy efficient as well as the fastest. This may assist in reducing travel times and eliminating the need for unnecessary delays (Parveen 2022). Travel applications or platforms powered by AI can give passengers customized help. They can provide real-time updates on traffic conditions, provide recommendations for the most appropriate mode of transportation for the user depending on the user’s preferences and the limits of the situation, and even help the user book tickets, lodgings, and other travel-related services (Fang 2019). AI can analyze demand patterns and make dynamic price adjustments to maximize resource allocation. For instance, ridesharing services may employ surge pricing during peak hours to motivate more drivers to be available, which will lower the time customers have to wait (Billhardt 2019).

2. Related works

This research investigates the feasibility of using artificial intelligence (AI) to improve the efficiency of public transportation systems and reduce traffic congestion. The results, which are the result of an in-depth review of previously published research, case studies, and the views of industry professionals, highlight numerous important areas in which AI might help the improvement of public transport systems. The constant monitoring of traffic patterns is made possible with the use of AI in real-time monitoring, which permits the correct forecast of congestion in the present moment (Kozlov 2022). The study produces insights regarding the manner in which AI may help build smarter cities and how it can do so in a variety of different ways. The methodological approach that will be used will be the research. The findings are organized according to the primary aspects of smart city improvement, including economics, society, atmosphere, and democracy (Yigitcanlar 2020). The study examines the many types and definitions of shared transportation services that are widespread in North America, and it summarizes the impact studies that have already been conducted. In addition, we investigate the confluence of shared mobility, electrification, and automation, including the possible consequences of shared automated vehicle (SAV) systems. Even though the effects of SAVs are not yet fully understood, several industry experts and university researchers anticipate increased productivity, decreased costs, and reduced emissions of greenhouse gases (Shaheen 2017). The exploration proposes consolidating four advancements: huge information, profound learning, in-memory registering, and design handling units (GPUs) to provide a complete way to deal with the speedy and versatile forecast of metro framework qualities. The primary aim of the research was to develop methods to expedite the creation of such forecasts (Aqib 2019). In this respect, although the use of artificial intelligence (AI) methods, including the fields of machine and deep learning, has attracted much interest in smart cities, less emphasis has been placed on the utilization of algorithmic optimization approaches. To be of assistance with this, this research gives an overview of strategies for optimization and software from the point of view of a smart city that is allowed by the Internet of Things (IoT) (Syed 2022). In general, obtaining all of the necessary data at each place is a fairly costly endeavor. In this research, an approach to deep learning is suggested for estimating the waiting time levels at transportation stations using proxy data and limited historical waiting time data at select stations. This approach is based on the idea that one can determine the waiting time levels using proxy data (Chu 2019). The study is an association of numerous problems that affect the transport sector and are categorized as autonomous vehicles. In regard to intelligent transport systems, some of the components that are being explored are connected to traffic management, public transport, safety management, manufacturing, and logistics. These are all areas where AI advantages are put to use (Iyer 2021).

3. Methodology

The closed-loop technique management system is shown in Figure 1 receives data about the median pixel-to-pixel matches of the standard mosaicked picture and the currently active mosaicked output from the comparator or mistake detectors. Infrared (IR), ultraviolet (UV), and inductive loop (IL) sensors have been replaced with numerous vision sensors in this modern take on control. BBNs, which employ membership processes and rule-based BBN sets controlled by problem detector data to make more informed decisions, are additionally employed to modify the onset of green and red lights in

https://www.malque.pub/ojs/index.php/msj
response to the density of road measurements. Vision equipment has been utilized to record pictures of vehicles for use in the feedback process. Multiple vision sensors are used so that precise distances between cars may be determined to provide reliable estimates of traffic volumes. After the photos have been taken, they go through a series of processes, including saving, grayscale conversion, transformation, wrapping, compositing, and eventually the production of the mosaicked picture.

A schematic representation of the suggested framework on a '+' shaped road is shown in Figure 2. Each side of the road now has four 12.0 megapixel webcams. It is expected that there are two lanes in each direction from west to east and east to west since this is where the majority of traffic flows. This is in contrast to the north-to-south and south-to-north directions, which only have one lane each owing to the relatively small volume of traffic in these directions. The suggested algorithm is designed to maintain the previously established schedule under typical traffic conditions. The algorithm determines the typical range for measuring traffic (which may be gleaned from historical data) and adjusts the timing of red and green lights appropriately. Because average density, average velocity, and average speed are normally the characteristics of a macroscopic traffic flow model, this proposed framework only accounts for the former.

3.1. Bayesian belief networks (BBNs)

The use of machine learning (ML) and deep learning (DL) algorithms to address practical issues has seen a meteoric rise in recent years. These algorithms have been used in several fields, such as medicine, electricity, picture identification, and indoor monitoring of objects. The suggested method relies on Bayes’ rule, which calculates the likelihood of an occurrence given prior knowledge of circumstances known to be relevant to that event. Bayes’ rule, which is based on conditional probability, is the bedrock of Bayesian inference. The statistical form of Bayes’ rule is as follows:

\[ b(Y|X) = \frac{b(Y|X)b(X)}{b(X)} \]  

(1)

The first term, commonly known as the prior probability, indicates an individual’s starting point of belief before any data concerning event Y is evaluated, whereas the second term may be thought of as a normalizing constant. If event Y has previously happened, then event X has a certain probability, denoted as \( b(Y|X) \). Since it is calculated after considering all available data, it is also referred to as the posterior probability of occurrence Y. The likelihood, or \( b(Y|X) \), measures how likely it is that event Y will occur given that event X has already taken place. The structure of a BBN and the probability distribution functions of its variables, known as the node parameters, characterize it. Given the nature of our investigation,
we restrict our attention to BBNs made up of discrete random variables. Bayesian networks (BBNs) are a crucial component of Bayesian inference; they are directed acyclic graphical models in which nodes represent random variables and arcs reflect causal relationships between the nodes. One-way links between BBN nodes suggest a tree-like or familial organization. Nodes’ familial relationships are shown in Figure 3, where W and Y, for example, represent parents and offspring, respectively. A node’s parent may have some effect on its offspring, but not vice versa. A node’s ancestors are all the nodes in the hierarchy above it, and a node’s descendants are all the nodes in the hierarchy below it. Finally, a node with no children is known as a sink node, while a node with no parents is known as a root node. Graphical representations of probabilistic models are useful because they allow for the concise and consistent depiction of joint probability functions. It is possible to obtain a more in-depth explanation of BBN architecture and its parts.

![Figure 3 Bayesian belief network illustration.](https://www.malque.pub/ojs/index.php/msj)

The parameters of the model also play a significant role in Bayesian inference by determining the node-specific conditional probability distribution (CPD). If we assume that the random variables are discrete, then we can use a conditional probability table (CPT) to depict the conditional connection between the nodes. BBN-based models vary in computing cost depending on their structure, node count, and state count for each variable. Performing probabilistic inference using BBNs is an NP-hard task for many researchers. If we assume that the two nodes X and Y in Figure 3 represent dichotomous random variables, then the resultant CPT will have $2^2$ states. It might be helpful to present some high-level ideas related to BBNs before continuing. The local Markov property states that a variable is conditionally independent of other variables while considering its immediate surroundings. Generalizing the local Markov property to BBNs is as follows:

$$Y_c \perp Y_{MT(c)}|Y_{BE(c)}$$  

If $Y_{MT(c)}$ is a nondescendant node, $Y_{BE(c)}$ is a parent node and $X_v$ is a random variable represented by a BBN node. Take the straightforward BBN shown in Figure 3 as an example; in this case, X is conditionally independent of the nondescendant $(U/X)$, which results in:

$$b(Y|U, X) = b(Y|U)$$  

The following equation provides a chain rule for decomposing a joint distribution of variables in a BBN.

$$b(Y_1, ..., Y_m) = b(Y_1|Y_1, ..., Y_{m-1}) \times b(Y_{m-1}|Y_1, ..., Y_{m-2}) \times (Y_2|Y_1)B(Y_1)$$  

As a further step, we may use equation 4 to obtain a generic version of the chain rule for BBN.

$$b(Y_1, ..., Y_m) = \prod_{j=1}^{M} b(Y_j|BE(b(Y_j)))$$  

### 3.1.1. Elimination of Variables Algorithm

It may be challenging to put BBNs into reality since real-world problems often include a large number of random variables, each of which might take on a large number of possible values. Inferring using BBNs is easy if you utilize the whole joint distribution and add up all latent variables. This is because the whole joint probability table for n binary variables will have $2^n$ entries, making the process tedious for large BBNs. To lessen the computing load during inference, a simple yet effective approach known as variable elimination (VA) may be implemented. We generalize below a scenario in which we need to determine a subset of the questioned variables Y using the evidence E and the latent variables X. The joint probability distribution of Y and E divided by the marginal probability distribution of E equals the conditional probability of Y giving evidence A.

$$b(Y|A = a) = \frac{b(Y|A = a)}{b(A = a)}$$  

To obtain the numerator of equation (6), marginalization over all latent variables $X_1,...,X_n$ is necessary.
\[ B(Y = y_j, A = a) = \sum_{x_1} \ldots \sum_{x_m} b(x_1, \ldots, x_m, Y = y_j, A = a) \quad (7) \]

As a means of avoiding unnecessary repetition of computations, we offer factors to serve as multidimensional tables. The collective probability of all variables may be written as a factorization. We may determine the joint probability of \( X \) and \( E \) by setting \( A_1 = a_1 \ldots A_r = a_r \) and marginalizing out the latent variables \( X_1, \ldots, X_r \) in turn.

\[ b(Y, A_1 = a_1, \ldots, A_r = a_r) = \sum_{x_1} \ldots \sum_{x_m} l(Y, A_1, \ldots, A_r, x_1, \ldots, x_m)_{A_1 = a_1, \ldots, A_r = a_r} \quad (8) \]

After that, we may write the product form of the joint factors using the chain rule for BBNs (equation (5)):

\begin{align*}
\frac{b(Y, BE(Y_j))}{l(Y, A_1 = a_1, \ldots, A_r = a_r)} &= l, b(Y, A_1 = a_1, \ldots, A_r = a_r) = \sum_{x_m} \ldots \sum_{x_1} l(Y, A_1, \ldots, A_r, x_1, \ldots, x_m)_{A_1 = a_1, \ldots, A_r = a_r} \\
&= \sum_{x_m} \ldots \sum_{x_1} \prod_{j=1}^r \{l_j\}_{A_1 = a_1, \ldots, A_r = a_r} \\
&= (9)
\end{align*}

As a result, solving the last term in equation (8), which is the sum of products, is equivalent to performing inference in BBNs. The nonlatent variable terms in equation (8) must be factored out before the final term can be effectively computed.

3.2. Standards and Limitations in Design

The following is assumed for developing the dynamic traffic light control system:

- Northbound, westbound, southbound, and eastbound traffic all converge at this isolated four-way intersection.
- The primary direction of travel is presumed to be east–west.
- There is no thought given to going right or left.
- The dynamic logic controller method will track the volume of traffic heading in both directions (north and south and west and east).
- The flow of traffic in one direction is interrupted by a lull in the other direction.

3.3. Vision Computing in (Mosaicking)

To create a mosaic picture, many photographs of the same subject are taken and then seamlessly stitched together. This process is essential for expanding the viewable area of a picture without degrading the quality of the final product. Digital photos are limited in size, so it is not always feasible to capture a whole region of interest. When this occurs, we may use image mosaicking to create a larger picture by combining many photos that intersect. The photographs are resampled and aligned with the coordinate system of one of the overlapping images to make a mosaic. For an image mosaicking system to work, it must account for factors including camera-to-camera distance, camera-to-scene distance, camera attributes, and scene content. Picture mosaics include many steps, some of which are shown in Figure 4. These steps include altering, mixing, and sewing images together.

![Image Processing Flow Diagram](https://www.malque.pub/ojs/index.php/msj)

**Figure 4** Image Processing Flow Diagram.

3.4. Cross-Correlation Pixel-to-Pixel Matched

Calculating the normalized cross-correlation coefficient between two templates reveals how closely their images resemble one another pixel by pixel. When the value of the normalized cross-correlation is higher than a predetermined cutoff, it is considered that a traffic light is located at the site in question. To provide the most accurate results, the template matching method relied on normalized cross correlation (sometimes called the distance measure or squared Euclidean distance).

\[ t_{i,d}^2(w, c) = \sum_{x,y}[(y, x) - d(y - w, x - c)]^2 \quad (10) \]
When feature $t$ is located in a window at coordinates $(u, v)$, the mosaicked image $(f)$ is added to the total over $(x, y)$ above the window. With $d^2$ growing, 

$$t^2(l, d)^2(w, c) = \sum_i(y, x) \equiv \left[(l^22)(y, x) - 2l(y, x)d(y - w, x - c) + d^22(y - w, x - c)\right]$$

If roughly constant, then there is no further cross-correlation term.

$$v(w, c) = \sum_{y, x} l(y, x) d(y - w - c)$$

$$y = \frac{\sum_{y, x} [l(y, x) - \overrightarrow{l_{w,c}}][d(y-w,x-c) - \overrightarrow{d}]}{\sum_{y, x} [l(y, x) - \overrightarrow{l_{w,c}}] \sum_{y, x} [d(y-w,x-c) - \overrightarrow{d}]}$$

The average value of $f(y, x)$ within the region defined by the template $d$ shifted to $(v, u)$ is denoted with $\overrightarrow{f_{u,v}}$ in Equation (14).

Equation (14) is preferable to other similar measures such as covariance or sum of absolute differences when calculating the matching degree because normalizing makes the computation more robust.

4. Results and Discussion

The effectiveness of the proposed framework was evaluated using a LabVIEW simulation test bed. There are vision sensors installed in the ‘+’ style of roads seen in the GUI that monitor the time of red and green lights to ensure that they are consistent. An experiment was conducted on one car to compare the suggested method to the standard method. On the Aim Sun test bed, one of the reference cars is permitted to travel from west to east at a speed of 40 kilometers per hour. The vehicle is timed for 450 seconds total, and the distance traveled is measured at the end of that period. Road performance is measured by imposing a random load on the road network. ANN, CNN, and SVM have all had their performances tested using traditional sensors under identical traffic circumstances. Figure 5 below provides a comparison of traditional methods with the suggested strategy. The results of the simulated experiment show that the vehicle travels farther when utilizing a BBN controller in conjunction with vision sensor computing (4.6 m in the allotted time frame vs. 4.3 m when using CNN, 4.0 m when using ANN, and 3.7 m when using SVM). The transportation infrastructure has been upgraded with the help of a new platform called Aim Sun. The efficiency of this system was tested. Additionally, the proportion of road traffic is shown. The average wait and travel times for cars are shown in the simulation results. The red light will turn on when no distance has been traveled and will stay on until traffic is present and the algorithm has been executed. During this time of inactivity, an algorithm determines when the following cycle will begin with a green light. The proposed framework (BBN) improves the accuracy percentage time while lowering the average waiting time, as shown in Figures 6 and 7 and Tables 1 and 2.

![Figure 5](image_url) Conventional methods compared to the suggested system.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM [21]</td>
<td>28</td>
</tr>
<tr>
<td>ANN [22]</td>
<td>37</td>
</tr>
<tr>
<td>CNN [23]</td>
<td>56</td>
</tr>
<tr>
<td>BBN [Proposed]</td>
<td>79</td>
</tr>
</tbody>
</table>
5. Conclusion

The purpose of this research is to build an algorithm that may be utilized to reduce traffic congestion on major thoroughfares in developing nations. We initially provide a simple picture mosaic technique for counting people in a busy area through live webcam feeds. We analyze real-time road traffic photos to demonstrate the occurrence and duration of congestion collapses. We hypothesize that localized decongestion transportation systems may enhance traffic flow at key nodes in road traffic networks and are possibly simpler to install in real-world scenarios. This prompts the creation of low-cost transportation methods to ease traffic in third-world nations. To optimize traffic flow and minimize average waiting time at a junction, a smart BBN controller solution using vision computing has been developed, which is predicated on precise measurements of the road's dynamic traffic density.

Ethical considerations

Not applicable.

Declaration of interest

The authors declare no conflicts of interest.

Funding

This research did not receive any financial support.
References


Iyer LS (2021) AI-enabled applications towards intelligent transportation. Transportation Engineering 5:100083.


