Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 6, June : 2024

# TRAFFIC FORECASTING FOR INTELLIGENT TRANSPORTATION SYSTEMS USING MACHINE LEARNING

**[1]Kakarla Ramcharan, [2]T. Chandra Sekhar**
[1]PG Scholar, Department of CSE, Srinivasa Institute of technology and science, Kadapa.
[2]Assistant professor, Department of CSE, Srinivasa Institute of technology and science, Kadapa.
[1]kakarlaramcharan2015@gmail.com, [2]chandu.nics@gmail.com

***ABSTRACT***

In this paper describing concept to control or automate green traffic signal allotment time based on congestion available at roadside using Canny Edge Detection Algorithm. Intelligent road traffic monitoring requires efficient systems to handle the vast amount of data generated by traffic surveillance cameras every second. Manual monitoring of this data is labor-intensive and impractical, necessitating the adoption of automated solutions. The primary objective of this work is to develop a fast and accurate traffic detection system that significantly reduces the need for human intervention. In this proposed work, we focus on accident detection and traffic flow analysis as key components of an intelligent road traffic monitoring system. The traffic surveillance data is pre-processed to construct a comprehensive training dataset. Using this dataset, we create a specialized CNN architecture by transferring a pre-trained network to traffic-related applications and retraining it with our self-established data. By utilizing the CNN, the system can effectively classify various multiclass problems, including accident detection, and identifying dense or sparse traffic conditions.

## 1. INTRODUCTION

Intelligent road traffic monitoring systems often integrate accident detection and traffic flow analysis into a unified platform. This integration allows for a holistic view of traffic conditions and enables more effective traffic management. For example, when an accident is detected, the system can automatically adjust traffic signals, reroute traffic, or provide alternative routes to minimize disruptions. It can also help emergency responders reach the scene faster by dynamically clearing traffic paths.

So, accident detection and traffic flow analysis are critical components of intelligent road traffic monitoring systems that leverage technology and data analysis to enhance road safety, reduce congestion, and optimize traffic flow. These systems play a pivotal role in improving transportation efficiency and overall quality of life for commuters by ensuring smoother traffic operations and quicker response to incidents on the road.

Traffic flow analysis, on the other hand, focuses on understanding and optimizing the movement of vehicles on the road. This involves collecting and processing vast amounts of data from various sources, including traffic cameras, GPS devices, and vehicle sensors. By analysing this data, traffic management systems can provide real-time information to drivers through digital signs, mobile apps, or navigation systems, helping them make informed decisions about their routes and reducing congestion. Traffic flow analysis also assists transportation agencies in making data-driven decisions about road design, infrastructure improvements, and traffic signal optimization to improve overall traffic efficiency.

## 2. LITERATURE SURVEY

**Lu et al. [3]** analyzed the location of a car in road transects, the road safety grade, the road surface condition, the visual condition, the vehicle condition, and the driver state were studied, and the prediction accuracy model of 86.67% was established.

**Alkheder et al. [4]** predicted the severity of traffic accidents from 16 attributes and four injury degrees (minor, moderate, severe, and death) through artificial neural networks.

**Akanbi et al. [5]** found that old age, overtaking, speeding, religious beliefs, poor braking performance, and bad tires were the main human factors causing and causing plant and animal extinctions in traffic accidents. Some effects of weather and accident conditions on the characteristics of highway traffic behaviour have also been pointed out by Caleffi et al.

**An et al. [7]** applied a fuzzy convolutional neural network to traffic flow prediction under uncertain traffic accident information and verified its effectiveness through the real trajectory of cars and meteorological data.

## 3. SYSTEM ANALYSIS

The Systems Development Life Cycle (SDLC), or Software Development Life Cycle in systems engineering, information systems and software engineering, is the process of creating or altering systems, and the models and methodologies that people use to develop these systems. In software engineering the SDLC concept underpins many kinds of software development methodologies.

### 3.1 Existing System

Manual Identification and Classification of the vehicle is a tedious process. So, Two Automatic Identification and Classification Algorithms namely Artificial Neural Network (ANN) and K - Nearest Neighbour (KNN) are used to detect and classify the vehicles using Acoustic signals which are recorded by microphones attached to the roads. The Manual prediction process has complexity in alone twofold narrow road with assorted traffic. This complexity is rectified by using the automatic detection algorithm (ANN & KNN). ANN & KNN is also used as a classifier and this broadly classifies the vehicles into three categories namely Heavy traffic, medium traffic and Low traffic. Microphones are used to detect the vehicles by recording the acoustic signals.

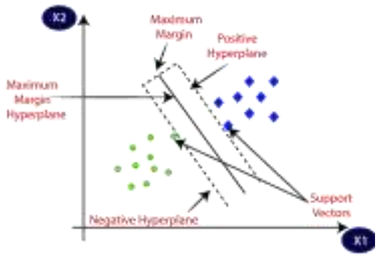



**Figure:** Analysis of SVM

It records the signal when a vehicle passes over the microphone and shows a peak in energy as a result of recording. The vehicle is automatically detected by locating the energy peak by smoothing the energy contour. The ANN/KNN classifiers are trained using the feature vectors. With the help of the Test data that contains over 180 vehicles, the efficiency of the method is tested with diverse categories.

### 3.2 Proposed System

The traffic surveillance system accumulates an enormous amount of data regarding road traffic each second. Monitoring these data with the human eye is a tedious process and it also requires manpower for monitoring. A deep learning approach can be utilized for traffic monitoring and control. The traffic surveillance data are pre-processed to construct the training dataset. The Traffic net is constructed by transferring the network to traffic applications and retraining it with a self-established data set. This Traffic net can be used for regional detection in large scale applications. Further, it can be implemented across-the-board. The efficiency is admirably verified through speedy discovery in the high accuracy in the case study. The tentative assessment could pull out to its successful application to a traffic surveillance system and has potential enrichment for the intelligent transport system in future.

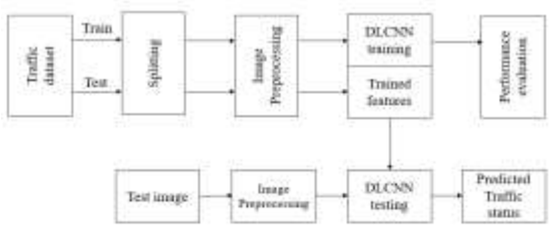



**Figure:** Proposed methodology

The block diagram of the proposed method, as shown in Figure outlines the various steps involved in the process of traffic status prediction using a Deep Learning Convolutional Neural Network (DLCNN) on the Traffic Net dataset.

### 3.3 Proposed DLCNN

Deep neural network is gradually applied to the identification of crop Traffic conditions and insect pests. Deep neural network is designed by imitating the structure of biological neural network, an artificial neural network to imitate the brain, using learnable parameters to replace the links between neurons. Convolutional neural network is one of the most widely used deep neural network structures, which is a branch of feed forward neural network. The success of DLCNN network model also confirms the importance of convolutional neural network model. Since then, convolutional neural networks have developed vigorously and have been widely used in financial supervision, text and speech recognition, smart home, medical diagnosis, and other fields.

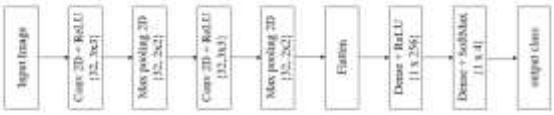



**Figure:** Proposed DLCNN

**Table:** Layers description

| Layer Names | No. of filters | Kernel size | Feature size |
|---|---|---|---|
| Conv 2D +ReLU | 32 | 3 x 3 | 62x62x32 |
| Max pooling 2D | - | 3 x 3 | 31x31x32 |
| Conv 2D+ReLU | 32 | 3 x 3 | 29x29x32 |
| Max pooling 2D | - | 3 x 3 | 14x14x32 |
| Flatten | - | 1x6272 | 1x6272 |
| Dense +ReLU | | 1 x 256 | 1 x 256 |
| Dense + SoftMax | | 1 x 4 | 1 x 4 |

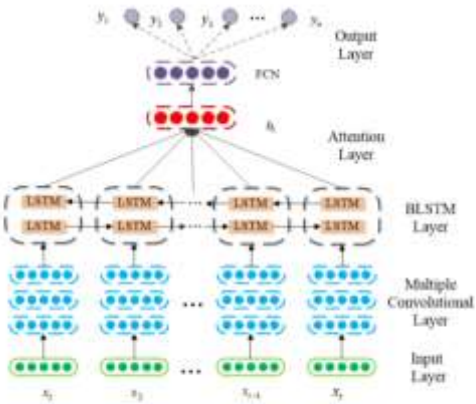
Convolutional neural networks are generally composed of three parts. Convolution layer for feature extraction. The convergence layer, also known as the pooling layer, is mainly used for feature selection. The number of parameters is reduced by reducing the number of features. The full connection layer carries out the summary and output of the characteristics. A convolution layer is consisting of a convolution process and a nonlinear activation function ReLU. A typical architecture of CNN model for crop Traffic condition recognition is shown above Figure.

### 2.4 Data Preprocessing

There are three symbolic data types in NSL-KDD data features: protocol type, flag and service. We use one-hot encoder mapping these features into binary vectors. One-Hot Processing: NSL-KDD dataset is processed by one-hot method to transform symbolic features into numerical features. For example, the second feature of the NSL-KDD data sample is protocol type. The protocol type has three values: tcp, udp, and icmp. One-hot method is processed into a binary code that can be recognized by a computer, where tcp is [1, 0, 0], udp is [0, 1, 0], and icmp is [0, 0, 1]

## 4. SYSTEM DESIGN

### 4.1 System architecture




**a) Employee**

Using employee module user who is working in company can register with application and login to application and view profile. Employee who want to check status or want to know if there are any anomaly is network connection or packets which are sent by him or any other employee inside company can request admin to check from the data set and update to employee. Employee can view anomaly packets inside the network after responding from admin.

**b) Admin**

Industrial Engineering Journal

ISSN: 0970-2555

Volume : 53, Issue 6, June : 2024

Admin a module who looks after network related issues in side a company who will login with application he can log dataset of networking packets which are part of the company and when ever he gets requests to check anomaly detection request admin will upload data set , preprocess data, divide data in to test and train and then create model and predict from the test data and predicted result is displayed to admin and sent to requested user.

**c) Dataset Collection**

In this step data set is collected from Kaggle website. Data set has features and labels. Features are used as input and labels for output.

**4.2 Data Preprocessing**

In this step data is preprocessed by removing unwanted data and NAN values and using features and labels which are useful to fit in to algorithm and then process data for prediction.

**a) Data split Test training**

In this stage data is divided in to test and train values using train test split function and store features and labels in to test train values. Train set is 30 percent of test set data which is used for checking accuracy of the dataset.

**b) Model Training**

In this stage different algorithms are used to check which algorithm provides best accuracy and select one algorithm to use that for fitting features and labels and then run algorithm in this way model is trained.

**c) Prediction and accuracy**

In this stage new input or test set is taken as input and given as input to predict function of the algorithm and then result of labels are as output of the algorithm.

**4.3 Input Design**

The input design is a component of the overall system design. The following is the main goal of the input design:

1. Create a low-cost input method.
2. To achieve the highest possible level of accuracy.
3. Ascertain that the user accepts and comprehends the input.

**4.4 Output Design**

Computer system outputs are primarily used to communicate the results of processing to users. They are also used to save a permanent copy of the results for future reference. In general, the various types of outputs are as follows:

- External Outputs, which have a destination outside of the organization
- Internal outputs have a destination within the organization and serve as the primary interface between the user and the computer.
- Operational outputs that are only used within the computer department.
- Interface outputs that engage the user in direct communication.

## 5. IMPLEMENTATION

**a) Design**

The software system design is produced from the results of the requirements phase. Architects have the ball in their court during this phase and this is the phase in which their focus lies. This is where the details on how the system will work is produced. Architecture, including hardware and software, communication, software design (UML is produced here) are all part of the deliverables of a design phase.

**b) Implementation**

Code is produced from the deliverables of the design phase during implementation, and this is the longest phase of the software development life cycle. For a developer, this is the main focus of the life cycle because this is where the code is produced. Implementation my overlap with both the design and testing phases. Many tools exists (CASE tools) to actually automate the production of code using information gathered and produced during the design phase.

## 6. TESTING

Testing is the process where the test data is prepared and is used for testing the modules individually and later the validation given for the fields. Then the system testing takes place which makes sure that all components of the system

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 6, June : 2024

property function as a unit. The test data should be chosen such that it passed through all possible condition. The following is the description of the testing strategies, which were carried out during the testing period.

During testing, the implementation is tested against the requirements to make sure that the product is actually solving the needs addressed and gathered during the requirements phase. Unit tests and system/acceptance tests are done during this phase. Unit tests act on a specific component of the system, while system tests act on the system as a whole.

So, in a nutshell, that is a very basic overview of the general software development life cycle model. Now let's delve into some of the traditional and widely used variations.

### 6.1 System Testing

Testing has become an integral part of any system or project especially in the field of information technology. The importance of testing is a method of justifying, if one is ready to move further, be it to be check if one is capable to with stand the rigors of a particular situation cannot be underplayed and that is why testing before development is so critical. When the software is developed before it is given to user to user the software must be tested whether it is solving the purpose for which it is developed. This testing involves various types through which one can ensure the software is reliable. The program was tested logically and pattern of execution of the program for a set of data are repeated. Thus the code was exhaustively checked for all possible correct data and the outcomes were also checked.

### 6.2 Module Testing

To locate errors, each module is tested individually. This enables us to detect error and correct it without affecting any other modules. Whenever the program is not satisfying the required function, it must be corrected to get the required result. Thus all the modules are individually tested from bottom up starting with the smallest and lowest modules and proceeding to the next level. Each module in the system is tested separately. For example the job classification module is tested separately. This module is tested with different job and its approximate execution time and the result of the test is compared with the results that are prepared manually. Each module in the system is tested separately. In this system the resource classification and job scheduling modules are tested separately and their corresponding results are obtained which reduces the process waiting time.

### 6.3 Integration Testing

After the module testing, the integration testing is applied. When linking the modules there may be chance for errors to occur, these errors are corrected by using this testing. In this system all modules are connected and tested. The testing results are very correct. Thus the mapping of jobs with resources is done correctly by the system.

### 6.4 Acceptance Testing

When that user fined no major problems with its accuracy, the system passers through a final acceptance test. This test confirms that the system needs the original goals, objectives and requirements established during analysis without actual execution which elimination wastage of time and money acceptance tests on the shoulders of users and management, it is finally acceptable and ready for the operation.

## 7. OUTPUT SCREENS

In this project author is using deep learning convolution2d neural network to detect traffic conditions such as heavy traffic, low traffic, accident or fire accident. To implement this project, we have used traffic net dataset available on GitHub. We have trained CNN model with traffic net dataset.

**Figure:** Home Page

**Figure:** trafficnet_dataset

**Figure:** Image Preprocessing

**Figure:** In above screen displaying sample image from processed images and now close above image to get below screen

**Figure:** In above screen CNN model generated with prediction accuracy as 99% and in below black console we can see CNN layer details

**Figure:** Test another image

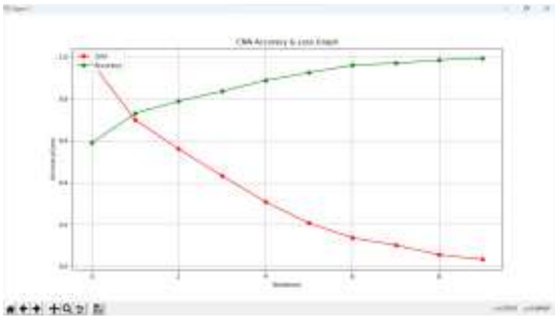


**Figure:** Accuracy & Loss Graph

In above graph red line indicate LOSS and green line indicates accuracy and x-axis represents number of iterations or EPOCH and y-axis represents accuracy/loss. In above graph we can see in each iteration accuracy get increase and loss/error rate decrease which means in every iteration CNN model get better.

## 8. CONCLUSION

In conclusion, the proposed method for traffic status prediction using DLCNN on the Traffic Net dataset presents a promising approach to addressing the challenges of traffic management, safety, and urban planning. This method harnesses the power of deep learning to accurately classify traffic conditions, offering advantages such as automation, scalability, and real-time monitoring. By providing precise insights into traffic dynamics, it enhances road safety, reduces congestion, and contributes to more efficient transportation systems. The model's adaptability and potential for continuous improvement make it a valuable tool for addressing the ever-evolving challenges of urban traffic. As cities continue to grow and traffic complexity increases, the significance of such methods cannot be overstated, offering a path towards smarter and safer urban mobility.

## REFERENCES

[1] K. G. Micheale, "Road traffic accident: human security perspective," International Journal of Peace and Development Studies, vol. 8, no. 2, pp. 15–24, 2017.

[2] T. K. Bahiru, D. K. Singh, and E. A. Tessfaw, "Comparative study on data mining classification algorithms for predicting road traffic accident severity," in Proceedings of the 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), IEEE, Coimbatore, India, September 2018.

[3] T. Lu, Z. H. U. Dunyao, Y. Lixin, and Z. Pan, "The traffic accident hotspot prediction: based on the logistic regression method," in Proceedings of the 2015 International Conference on Transportation Information and Safety (ICTIS), IEEE, Wuhan, China, June 2015.

[4] S. Alkheder, M. Taamneh, and S. Taamneh, "Severity prediction of traffic accident using an artificial neural network," Journal of Forecasting, vol. 36, no. 1, pp. 100–108, 2017.

[5] G. Akanbi, O. E. Charles-Owaba, and A. E. Oluleye, "Human factors in traffic accidents in Lagos, Nigeria," Disaster Prevention and Management, vol. 18, no. 4, pp. 397–409, 2009.

[6] F. Caleffi, S. T. Lucchesi, M. J. Anzanello, and H. B. B. Cybis, "Influência das condições climáticas e de acidentes na caracterização do comportamento do tráfego em rodovias," Transport, vol. 24, no. 4, pp. 57–63, 2016.

[7] J. An, L. Fu, M. Hu, W. Chen, and J. Zhan, "A novel fuzzy-based convolutional neural network method to traffic flow prediction with uncertain traffic accident information," IEEE Access, vol. 7, pp. 20708–20722, 2019.

[8] S. H. A. Hashmienejad and S. M. H. Hasheminejad, "Traffic accident severity prediction using a novel multi-objective genetic algorithm," International Journal of Crashworthiness, vol. 22, no. 4, pp. 425–440, 2017.

[9] H. Ren, Y. Song, J. Wang, Y. Hu, and J. Lei, "A deep learning approach to the prediction of short-term traffic accident risk," 2017, https://arxiv.org/abs/1710.09543.

[10] H. Ren, Y. Song, J. Wang, Y. Hu, and J. Lei, "A deep learning approach to the citywide traffic accident risk prediction," in Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), IEEE, Maui, HI, USA, November 2018.

[11] C. Huang, C. Zhang, P. Dai, and L. Bo, "Deep dynamic fusion network for traffic accident forecasting," in Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, November 2019.

[12] B. Kumeda, F. Zhang, F. Zhou, S. Hussain, A. Almasri, and M. Assefa, "Classification of road traffic accident data using machine learning algorithms," in Proceedings of the 2019 IEEE 11th international conference on communication software and networks (ICCSN), pp. 682–687, IEEE, Chongqing, China, June 2019.

[13] Y. L. Murphey, K. Wang, L. J. Molnar et al., "Development of data mining methodologies to advance knowledge of driver behaviors in naturalistic driving," SAE International journal of transportation safety, vol. 8, no. 2, pp. 77–94, 2020.

[14] W. Bao, Y. Qi, and Y. Kong, "Uncertainty-

based traffic accident anticipation with spatio-temporal relational learning," in Proceedings of the 28th ACM International Conference on Multimedia, Seattle, WA, USA, October 2020.

[15] T. T. Yaman, E. Bilgic, and M. Fevzi Esen, "Analysis of traffic accidents with fuzzy and crisp data mining techniques to identify factors affecting injury severity," Journal of Intelligent and Fuzzy Systems, vol. 42, no. 1, pp. 575–592, 2022.

[16] C. Iwendi, A. K. Bashir, A. Peshkar et al., "COVID-19 patient health prediction using boosted random forest algorithm," Frontiers in Public Health, vol. 8, 2020.

[17] J. Ding, Q. Dai, W. Fan et al., "Impacts of meteorology and precursor emission change on O3 variation in Tianjin, China from 2015 to 2021," Journal of Environmental Sciences, vol. 126, pp. 506–516, 2023.

[18] M. Schonlau and R. Y. Zou, "The random forest algorithm for statistical learning," STATA Journal, vol. 20, no. 1, pp. 3–29, 2020.

[19] M. Yan and Y. Shen, "Traffic accident severity prediction based on random forest," Sustainability, vol. 14, no. 3, 2022.

[20] H. Zhao, X. Li, H. Cheng, J. Zhang, Q. Wang, and H. Zhu, "Deep learning-based prediction of traffic accidents risk for Internet of vehicles," China Communications, vol. 19, no. 2, pp. 214–224, 2022.

[21] M.-M. Chen and M.-C. Chen, "Modeling road accident severity with comparisons of logistic regression, decision tree and random forest," Information, vol. 11, no. 5, 2020.

[22] H. Koma, T. Harada, A. Yoshizawa, and H. Iwasaki, "Detecting cognitive distraction using random forest by considering eye movement type," International Journal of Cognitive Informatics and Natural Intelligence, vol. 11, no. 1, pp. 16–28, 2017.

[24] L. Wang, J. Wu, R. Li et al., "A weight assignment algorithm for incomplete traffic information road based on fuzzy random forest method," Symmetry, vol. 13, no. 9, 2021.

[25] Y. Zhang, H. Li, and G. Ren, "Estimating heterogeneous treatment effects in road safety analysis using generalized random forests," Accident Analysis & Prevention, vol. 165, Article ID 106507, 2022.

[26] J. Gan, L. Li, D. Zhang, Z. Yi, and Q. Xiang, "An alternative method for traffic accident severity prediction: using deep forests algorithm," Journal of Advanced Transportation, vol. 2020, Article ID 1257627, 13 pages, 2020.