ACTA TECHNICA CORVINIENSIS – Bulletin of Engineering | e–ISSN: 2067 – 3809
Tome XVIII [2025] | Fascicule 2 [April – June]

[1.] Ayush Kumar Ruplal BURDE, [1.] Abhinav Vijay PATIL, [1.] Durvesh Satish PATIL, [1.] T.P. RAJU

DEEP LEARNING FOR REAL–TIME VEHICLE COUNTING AND CATEGORIZATION IN TRAFFIC VIDEOS

[1.] Tulsiramji Gaikwad Patil College Of Engineering, Nagpur, INDIA

Abstract: Efficient traffic monitoring has become a critical need in today's rapidly expanding urban environments. Traditional methods using physical sensors to detect and classify vehicles are not only expensive but also require regular maintenance. To address these limitations, this project proposes a vision–based traffic analysis system that is both cost–effective and efficient. The system works by extracting frames from video footage and using the Gaussian Mixture Model (GMM) for background subtraction to detect and count vehicles. After detection, vehicles are classified by comparing their contour areas with predefined size values. A major contribution of this project is the comparison between two classification techniques: Contour Comparison (CC) and Bag of Features (BoF). Both methods are analyzed to determine their accuracy in identifying different types of vehicles. This approach offers a scalable and affordable solution for traffic analysis, eliminating the need for costly hardware while still delivering reliable performance in real–time traffic environments.

Keywords: Vehicle counting, Business analysis, Contour Comparison

INTRODUCTION

In today's world, there is a growing demand for secure and cost–effective systems to automate vehicles and reduce vehicle theft. With increasing traffic on roads and highways, along with limitations in current vehicle detection methods, new technologies are being developed to enhance vehicle identification. Among these, computer vision systems have emerged as a popular solution, though they still face several challenges when it comes to accurate classification. Intelligent vision systems that can detect and track moving vehicles in real time are crucial for many technological and research applications. These systems help extract valuable information such as traffic flow, vehicle speed, driving behaviour, and traffic trends. Relying on manual monitoring is no longer practical. Therefore, the development of smart systems capable of extracting traffic and vehicle classification data is vital for modern traffic management. Additionally, such systems play a key role in accident analysis, where vision technology helps detect and classify vehicles involved in recorded incidents. A figure serves as a visual illustration of such data.

As the demand for smarter transportation systems grows, there is an urgent need for secure and budget–friendly solutions to automate vehicle monitoring and prevent theft. Rising traffic volumes on roads and highways, along with inefficiencies in traditional vehicle detection methods, have driven the development of advanced detection technologies. Computer vision has become a leading choice in this area, although it still presents challenges in achieving accurate vehicle classification. Real–time tracking and detection of moving vehicles by intelligent vision systems are essential for a wide range of technological and research–based applications. These systems allow the extraction of important data such as traffic density, vehicle speed, driver behaviour, and traffic trends. Manual monitoring is becoming outdated, making the creation of intelligent systems capable of analysing and classifying traffic information increasingly important. Furthermore, such systems are valuable in accident investigations, where vision–based tools help identify and classify vehicles involved. A figure is used as a visual representation of this data. Pix may be divided into the posterior three types. A double picture consists of pixels that may be considered one of colours, generally black and white. Double snap shots are called situations or two situations. This way that each pixel is stored as an integer, i.e. 0 or 1. Gray is an intermediate colour among white and white. It's a unprejudiced shade or achromatic colour, which actually means" white" color because it's suitable to be composed of white and white. It came the achromatism of cloudy skies, dust and lead. An achromatism (virtual) print is a virtual picture that includes colour information for every pixel. This fashion is environmentally affable as it does not suffer chemical processing. Digital imaging is constantly used to validate and report literal, clinical, and specific events This paper describes an imaginative and visionary– rested device for detecting, covering and classifying shifting motorcars. Four different implicit enterprise pots can be described, but the proposed software program is lissom and the wide variety of agencies

LITERATURE SURVEY

Improving road condition analysis for traffic control and safety has become a key focus in intelligent transportation research. Alpatov et al. tackled this issue by developing image processing algorithms for vehicle detection, counting, and road sign recognition, all optimized for use with stationary camera images. These advanced methods were also tested on embedded smart camera systems to enhance performance. In a related effort, Singh and colleagues introduced a vision–based vehicle identification and counting system. Their contribution includes a high–level annotated dataset containing 57,290 annotations across 11,129 images. Unlike many public datasets, their collection features small, reflective objects, offering a more robust foundation for deep learning–based vehicle detection. Neupane et al. also contributed valuable advancements in this area, further supporting the progress of intelligent traffic monitoring systems. To spoil P2, exceptional– tuning is grounded on this skilled and applied transfer workout in an ultramodern YOLO (You Only Look previously) network. For P3, this paintings proposes a multivehicle shadowing set of rules that snappily calculates every path, classifies, and obtains the vehicle haste. Lin et al. It introduces a business tracking device grounded on digital discovery zones, Gaussian admixture model (GMM) and YOLO to ameliorate automobile counting and bracket effectiveness. GMM and virtual discovery bands are used to rely cars and YOLO is used to classify vehicles. Additionally, car distance and time are used to estimate automobile velocity. In this take a look at, Montevideo Audio and Video Data (MAVD), GARM Road– Traffic Monitoring Dataset (GRAM– RTM) and our series dataset are used to check the proposed gadget. Chauhan et al. It makes use of a complicated Convolutional Neural Network (CNN) item. This work evaluates the quiescence, energy and tackle costs of planting exploration grounded on our CNN model, as the growing region lacks strong network connectivity for nonstop videotape streaming from the road to the pall garçon.

As intelligent traffic systems continue to evolve, Arinaldi et al. introduced a computer vision–based video analysis system aimed at automating the collection of key traffic statistics. The system provides data such as vehicle counts, type classification, and speed estimation from video footage, helping policymakers and traffic regulators make informed decisions. To achieve this, two models were developed: one based on a MoG–SVM network and the other utilizing Faster R–CNN, a leading deep learning architecture for object detection. In a similar direction, Goma et al. proposed a real–time vehicle detection and counting approach using YOLOv2, enhanced by motion–based feature analysis for greater accuracy. This work is grounded on the discovery of coetaneous vehicle features and tracking to achieve accurate counting results. The proposed strategy works in two phases; the first is to identify vehicles and the second is to count moving vehicles. For original object discovery, this work uses the fastest literacy object discovery algorithm YOLOv2 before filtering K– groups and KLT shamus. An effective approach is also introduced using the temporal information of the inter–frame discovery and shadowing features to marker and directly calculate each vehicle with its separate line. Oltean et al. It proposes a real– time vehicle counting approach using a small YOLO for shadowing. This program works on Ubuntu with GPU processing and the coming step is to test it on low budget bias like the Jetson Nano. Test results show that this approach achieves high delicacy in real– time speed (33.5 FPS) in real business videotape.

Pico et al. proposed to apply a low– cost system for vehicle identification and bracket using an ARM– grounded platform (ODROID XU– 4) installed with the Ubuntu operating system. The algorithm used is grounded on an open source library (Intel OpenCV) and is enforced in the Python programming language. Trials prove that the effectiveness of the enforced algorithm is 95.35, but it can be bettered by adding the training samples. Tituana et al. review colourful former workshop developed in this field and identify the technological styles and tools used in those workshop; In addition, this study also highlights trends in this field. The most applicable papers are reviewed and the results are epitomized in tables and numbers.

PROPOSED SYSTEM

This system can be used to descry, fete and track vehicles in videotape images, and also classify the detected vehicles into three different classes grounded on their size. The proposed system is grounded on three modules, videlicet background literacy, focus birth, and vehicle bracket as described in background deduction, a classic approach to capture background images or in other words, moving object discovery.

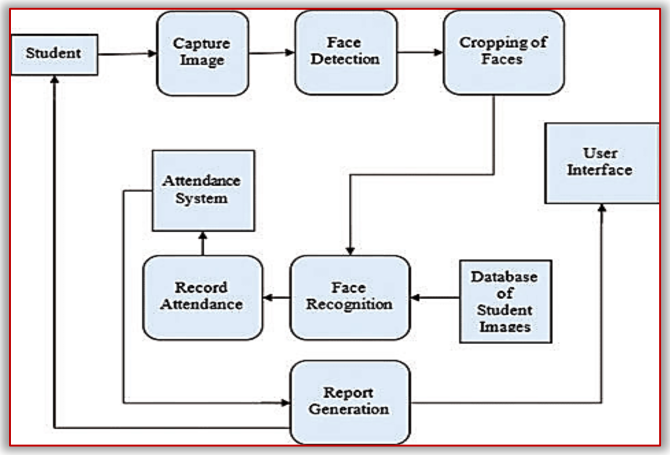

Figure 1: Block diagram of proposed system.

Vehicle Detection and Counting

The final stage of the proposed system is vehicle classification. Once the pre–processing module is applied, silhouettes of the detected objects are obtained, and key shape features such as centroids, aspect ratio, area, size, and compactness are extracted for classification purposes. This module operates through three main steps: background subtraction, image enhancement, and feature extraction. Background removal is performed by setting the static pixels of stationary objects to zero, helping isolate the moving objects. Following this, processes like noise filtering, dilation, and erosion are applied to refine the object silhouettes.

The output of this module provides a clear identification of the region of interest in the initial frame of the video, I draw a line near the image and define the ROI. The thing is to fete ROI in after frames, but fete that ROI is n't the primary vehicle. This is only part of the vehicle and can be misshaped, rotate, restate, or indeed fully vanish from the frame. Vehicle discovery A visionary strategy for opting a hunt window for vehicle discovery using image environment, a GMM frame is proposed for vehicle discovery with successional movement with top–down attention, constantly achieved satisfactory performance in relating vehicles clear link box. Proposed a methodical hunt strategy for detecting visual vehicles in the image, where the discovery model proposed a deep RL frame to elect the applicable action to capture the vehicle in the image. Vehicle Count This module counts detected vehicles and the result of this count will be streamlined constantly grounded on vehicle discovery, the result will be affair for streaming videotape using OpenCV.

Background Learning Module

The first module in this system is to learn how the background differs from the focus. Also, since the proposed system works on the videotape feed, this module excerpts frames from it and learns about the background. In a business scene captured by a stationary camera mounted on the roadside, moving objects can be considered focus and stationary objects can be considered background. An image processing algorithm is used to learn about the background using the system described over.

Gaussian Mixture Modelling (GMM)

At its core, the Gaussian Mixture Model (GMM) is a clustering technique that models data using multiple Gaussian distributions. Unlike hard clustering methods like k–means, GMM allows for soft assignments—meaning each data point can belong to more than one cluster with certain probabilities. Each Gaussian component in the model represents a group and holds a specific "responsibility" for generating particular data points, offering a more flexible and probabilistic approach to data grouping.

How can we estimate the appearance of this model? Well, one thing we can do is introduce retired variables Data (gamma) for each data point. This assumes that each data point is generated using some information about the idle variable. In other words, it tells you that the Gaussian generated a certain data point. But in practice we do not see these retired variables, so we've to estimate them. How to do this? Well, luckily for us, we formerly have an algorithm that works in this kind of situation, the Expectation Maximization (EM) algorithm, and we'll talk about it next.

The EM algorithm

The EM algorithm consists of two way, theE–step or Anticipation step and the M– step or

Maximization step. Suppose we've some idle variables (unobserved and defined by the vector Z below) and our data point X. Our thing is to maximize the borderline probability of X given our parameters(defined by the vector). In fact, we can find the borderline distribution as the union of X and Z and find the sum of all Z's(rule of probability).

$$\ln p(X \mid \Theta) = \ln\{\Sigma = (X.Z|\theta)\}$$

The above equations frequently produce complex functions that are delicate to gauge. What we can do in this case is to use Jensens Inequality to construct a lower set function that's easier to optimize. However, we can compare the original function, if we optimize this by minimizing the KL difference (gap) between the two distributions. This process is described. I've also handed a link to a videotape showing the derivate of the KL divergence for those who want a more rigorous fine explanation.

In fact, we only need to do two way to estimate our model. In the first step (E–step), we want to estimate the posterior distribution of our idle variable in terms of tentative weight (π), our term (μ) and the Gaussian mean covariance (Σ). Also we can enter the alternate step (M–step) and use it Increase the liability associated with our parameter parameter θ. This process is repeated until the algorithm converges (the loss function remains unchanged).

Bag of Features Model

Model Visual Features (BOF) is one of the most important generalities in computer vision. We use a visual vocabulary model to classify image content. It's used to make a high volume shadowing system (non–specific, precise). When we classify textures using textures, we rather use a visual vocabulary model. As the name suggests, the conception of" visual bag of words" is actually deduced from the" bag of words" model used in information reclamation (eg, textbook– grounded hunt machines) and textbook analysis. The general idea in Word Bag is to present a" document"(ie a web runner, a Word train, etc.) as a collection of important keywords, fully ignoring the order in which the words appear. Documents that partake the same keyword are considered to belong to each other anyhow of the order of the keywords. Also, because we fully ignore the order of words in a document, we call this representation" bag of words" rather of" list of words" or" list of words" Treating a document as a" bag of words" allows us to efficiently dissect and compare documents. because we do not need to store information about the order or position of words– we count how numerous times a word appears in a document, and also use the frequency number for each word as a way to rate the document. In computer vision, we can use the same conception– only now rather of working with keywords, our" words" are now layers of images and related point vectors.

Given a wordbook of possible visual words, we can also count the number of times each visual word appears and fantasize it as a histogram. This histogram is a veritable bag of visual words. Structure visual vocabulary can be divided into three way.

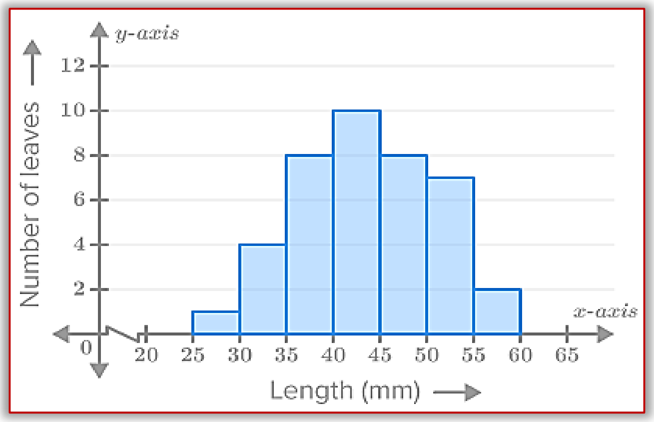

Figure 2. Length of the leaves

— **Step # 1:** point birth

The first step in erecting a visual bag of words is to prize descriptors and features from each image in our database. point birth can be done in several ways identify crucial points and excerpt SIFT features from crucial regions of our image; using circles in regular intervals(for illustration, solid button sensors) and derivations of other forms of original steady descriptors; or we can prize the average RGB value from a arbitrary image region. The point then's that for each input image we get several point vectors

— **Step # 2:** Dictionary/ Vocabulary Construction

After rooting point vectors from each image in our database, we need to make a vocabulary of possible visual words. Word conformation is generally done through a k– means clustering algorithm that summarizes the point vectors attained from step 1. The centers of the performing clusters (eg, centroids) are considered as visual word wordbooks.

— **Step #3:** Vector Quantization

Given an arbitrary image (whether from our original database or not), we can identify and abstract the image using a bag of image words using this process prize the point vector as in step 1 over. For each uprooted point vector, count its nearest neighbours in the wordbook created in step 2– this is generally done using Euclidean distance. Take the set of nearest neighbour markers and construct a histogram of length k (the number of clusters formed by kverbs), where the i value in the histogram is the frequency of the i–visual word. When modelling an object by distributing prototype vectors, this process is generally called vector quantization.

Classification

An interesting aspect of our network is its simplicity—the classifier is replaced with just a masking layer, without relying on any prior or convolutional structures. However, to achieve accurate performance, the model must be trained on a large and diverse dataset, ensuring vehicles of various sizes and positions are well represented throughout the training process. Visual shadowing solves the problem of chancing a target in a new frame from the current position. The proposed shamus stoutly tracks the target with sequence movement controlled by GMM. GMM predicts the stir to run the target from its position in the former frame. The crossroad box moves with the movement prognosticated from the former state, and the coming movement continues to be prognosticated from the moved state. We break the vehicle shadowing problem by repeating this process in a series of tests. GMM excels in RL as well as SL. Online adaption takes place during real shadowing.

A GMM is designed to induce stir to find the position and size of the target vehicle in a new frame. The GMM algorithm learns a policy that selects the most optimal action to follow from the current situation. In GMM, a policy system is developed in which the input is an abbreviated image subcase in the former state and the affair is the probability distribution of conduct similar as restatement and scale change.

4. RESULTS

The process of choosing this course of action requires a bit of exploration step more than the sliding window or seeker slice approach. likewise, since our system can localize the target by opting the stir, post–processing similar as box retrogression is not needed.

Advantages of Proposed System

- Identify high– moving vehicles in videotape sequences.
- Vehicle shadowing is uncorked.
- Identify the type of vehicle.
- Calculate the quantum of business through videotape.

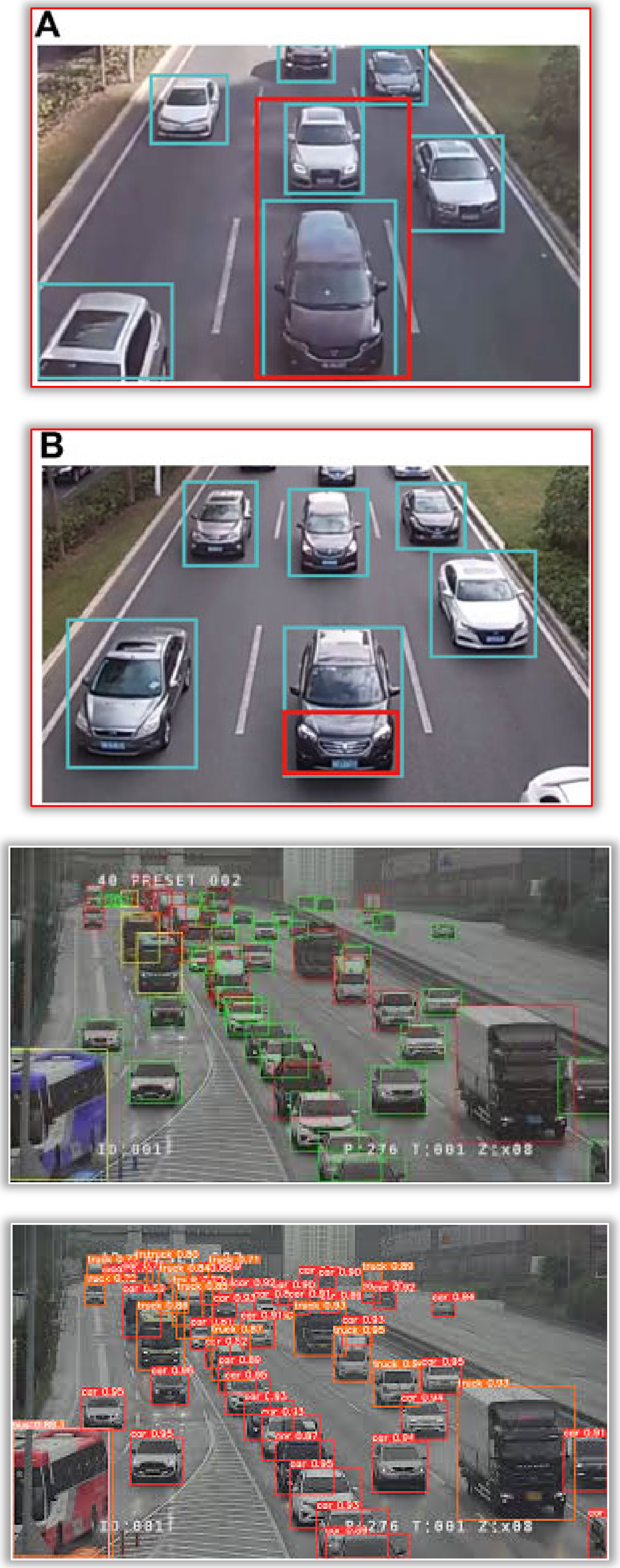

Figure 3. Results

CONCLUSION AND FUTURE SCOPE

The proposed system is developed in Python using OpenCV libraries, processing camera images from various sources. A user–friendly interface allows users to select regions of interest for analysis, while image processing techniques are applied to detect and count vehicles. Machine learning algorithms are then used for vehicle classification. Experimental results show that the Contour Comparison (CC) method consistently outperforms both the Bag of Features (BoF) and SVM methods, delivering classification results that are more closely aligned with ground truth values.

A notable drawback of the system is its limited ability to detect occluded vehicles, which directly affects the accuracy of vehicle counting and classification. This challenge can be addressed by incorporating secondary classification methods, such as colour–based techniques. Another limitation lies in the need for manual input to define regions of interest—users must draw an imaginary line through the centre of vehicle contours, making the system reliant on human judgment. Additionally, variations in camera angles can impact performance, but this can be improved using camera calibration techniques to optimize road visibility. The system also faces difficulties in night–time detection, as it depends on clear visibility for contour and SIFT feature extraction. To enhance accuracy, the system can be further refined using advanced image segmentation and artificial intelligence methods.

References

[1] Chauhan, M. S., Singh, A., Khemka, M., Prateek, A., & Sen, Rijurekha. (2019). Bedded CNN–grounded vehicle bracket and counting in non–laned road business. Proceedings of the Tenth International Conference on Information and Communication Technologies and Development (ICTD '19), Article 5, 1–11.

[2] Butt, M. A., Khattak, A. M., Shafique, S., Hayat, B., Abid, S., Kim, Ki–Il, Ayub, M. W., Sajid, A., & Adnan, A. (2021). Convolutional neural network grounded vehicle bracket in adverse luminous conditions for intelligent transportation systems. Complexity, Article ID 6644861, 11 pages.

[3] Neupane, Bipul, et al. (2022). Real–time vehicle bracket and tracking using a transfer literacy–bettered deep learning network. Sensors (Basel, Switzerland), 22(10), 3813.

[4] Gomaa, A., Minematsu, T., Abdelwahab, M. M., et al. (2022). Faster CNN–grounded vehicle discovery and counting strategy for fixed camera scenes. Multimedia Tools and Applications, 81, 25443–25471

[5] Jahan, N., Islam, S., & Foysal, M. F. A. (2020). Real–time vehicle bracket using CNN. 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 1–6.

[6] Lin, C. J., Jeng, Shiou–Yun, & Lioa, Hong–Wei. (2021). A real–time vehicle counting, speed estimation, and bracket system grounded on virtual discovery zone and YOLO. Mathematical Problems in Engineering, Article ID 1577614

[7] Oltean, G., Florea, C., Orghidan, R., & Oltean, V. (2019). Towards real–time vehicle counting using YOLO–bitsy and fast stir estimation. 2019 IEEE 25th International Symposium for Design and Technology in Electronic Packaging (SIITME), 240–243.

[8] Pico, L. C., & Benítez, D. S. (2018). A low–cost real–time bedded vehicle counting and bracket system for traffic management applications. 2018 IEEE Colombian Conference on Communications and Computing (COLCOM), 1–6.

[9] Balid, W., Tafish, H., & Refai, H. H. (2018). Intelligent vehicle counting and bracket detector for real–time business surveillance. IEEE Transactions on Intelligent Transportation Systems, 19(6), 1784–1794

[10] Alpatov, Boris, Babayan, Pavel, & Ershov, Maksim. (2018). Vehicle discovery and counting system for real–time business surveillance. 1–4

[11] Tituana, D. E. V., Yoo, S. G., & Andrade, R. O. (2022). Vehicle counting using computer vision: A survey. 2022 IEEE International Conference for Confluence in Technology (I2CT), 1–8.

[12] Song, H., Liang, H., & Li, H., et al. (2019). Vision–grounded vehicle discovery and counting system using deep literacy in trace scenes. European Transport Research Review, 11, 51

[13] Arinaldi, A., Pradana, J. A., & Gurusinga, A. A. (2018). Discovery and bracket of vehicles for business videotape analytics. Procedia Computer Science, 144, 259–268.

[14] Khan, A., Sabeenian, R. S., Janani, A. S., & Akash, P. (2022). Vehicle bracket and counting from surveillance camera using computer vision. In Suma, V., Baig, Z., Shanmugam, S. K., & Lorenz, P. (Eds.), Inventive Systems and Control. Lecture Notes in Networks and Systems, vol. 436. Springer, Singapore.

[15] Gonzalez, P., & Nuño–Maganda, M. A. (2014). Computer vision–grounded real–time vehicle shadowing and bracket system. Midwest Symposium on Circuits and Systems, 679–682.

ISSN: 2067–3809