Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

DeepTraff: Real-Time Traffic Sensing and Vehicle Category Analytics using Deep Learning

Vibhor Kumar Vishnoi[1], Anshika Yadav[2], Akshita[3], Abhishek Kumar[4], Ananya Agarwal[5],

[1, 2, 3, 4, 5] **Department of Computer Science & Engineering, Moradabad Institute of Technology, Moradabad-244001, Uttar Pradesh, India.**

(Corresponding Author: [1]anshikayadav7604@gmail.com)

***Abstract*—**Rapid urbanization in India and other developing countries has driven an explosive rise in vehicle ownership and heterogeneous traffic flows. This growth has outpaced traditional traffic monitoring methods, which depend heavily on static sensors, manual observation, and basic CCTV usage without intelligent analysis. These legacy approaches provide limited real-time insight and almost no category-wise understanding of traffic composition, making them unsuitable for modern smart- city applications. This paper presents *DeepTraff*, a conceptual real-time traffic sensing and vehicle analytics framework that leverages deep learning-based object detection model to provide vehicle analytics in complex urban conditions. It utilizes YOLO algorithm that operates on live CCTV video feeds and classifies vehicles into categories such as two-wheelers, three-wheelers, and four-wheelers, which are typical of Indian roads. In addition, the model provides analysis on vehicle count and peak traffic time. A modular backend aggregates the model output into interpretable statistics, and a web-based dashboard visualizes these metrics for planners, administrators, and traffic police. Without relying on explicit mathematical modeling or algorithm listings, *DeepTraff* emphasizes pragmatic, deployable ideas that can be integrated into city-level infrastructures. The framework illustrates how camera-based AI analytics can support congestion assessment, lane planning, policy evaluation, and future extensions such as speed monitoring and violation detection. Although this work does not present experimental numerical results, it lays a confident, implementation-oriented foundation for subsequent real-world deployments and academic research.

Keywords— Deep Learning, YOLO, Traffic Analytics, Smart Cities, Heterogeneous Traffic, Real-Time Surveillance, Vehicle Categorization.

I. INTRODUCTION

Nowadays, in India many cities face challenges such as escalating vehicle density and highly heterogeneous traffic behavior. Indian urban roads typically accommodate motorcycles, scooters, auto-rickshaws, private cars, e-rickshaws, mini-vans, and minibuses simultaneously within the same lane which likely not seen in many western road networks that feature lane discipline, homogeneous vehicle types, and structured driving patterns. This diverse behavior of traffic in Indian cities makes traffic behavior complex, non-linear, and

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

difficult to model using conventional techniques. Despite the increasing deployment of CCTV networks under smart-city initiatives, most of these systems continue to function merely as recording devices or as passive surveillance tools monitored by human operators. In several control rooms, multiple screens are observed simultaneously, yet the footage rarely contributes to structured analytics or real-time insights. Legacy monitoring techniques include ground-embedded sensors, handheld or stationary radar devices, manually conducted traffic counts, and basic CCTV feeds that typically require continuous human supervision. Although such tools can support limited monitoring tasks, they are inadequate for capturing de- tailed traffic composition or responding proactively to rapidly changing urban mobility conditions.

At the same time, policymakers are increasingly interested in granular traffic details. Questions such as "What proportion of traffic at this junction consists of two-wheelers?", "How many auto-rickshaws operate during peak hours?", or "Which corridors are dominated by private cars?" are central for defining bus lanes, encouraging public transport, designing parking infrastructure, and formulating emission policies. Unfortunately, traditional tools rarely answer these questions. Advances in deep learning and computer vision have opened new possibilities for answering such questions in real time. Object detection models such as **YOLO [1]–[6] (You Only Look Once)** can automatically identify and localize vehicles in video frames, making it possible to convert raw CCTV feeds into structured data. Coupled with modern backend frameworks and web dashboards, this enables the construction of end-to-end systems for live traffic analytics.

DeepTraff is conceptualized in this context. It proposes a practical architecture that fuses deep learning-based perception with configurable analytics and visualization. The system focuses on categorizing vehicles into meaningful classes that reflect Indian road realities, and on presenting these insights in a form that administrators can use directly. The remainder of this paper elaborates on the related work, problem formulation, proposed architecture, dataset and annotation, deployment considerations, smart-city integration, limitations, and future research directions.

II. Literature Review

A wide variety of research has explored traffic monitoring using computer vision and machine learning. Early work centered on background subtraction to isolate moving objects, followed by blob tracking to estimate vehicle counts. Such approaches are highly sensitive to environmental variations, and their performance deteriorates in dense or mixed traffic, where shadows, partial occlusions, and lighting inconsistencies are common.

The emergence of deep learning brought major improvements. Convolutional Neural Networks (CNNs) [13], [14] became the primary tool for image-based perception tasks. Two-stage detectors like Faster R-CNN [15]–[17] demonstrated high accuracy by first proposing candidate regions and then classifying them. However,

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

their computational costs limited usage in real-time, multi-camera scenarios. Single- stage detectors such as SSD and the YOLO [1]–[6] family addressed this by predicting object locations and categories in a single forward pass, enabling higher frame rates.

Multiple authors have applied deep detectors to traffic scenes. Studies have reported vehicle counting on highways, estimation of queue lengths at signals, and detection of illegal lane usage. Others have integrated Automatic Number Plate Recognition (ANPR) with detection to build enforcement systems for speeding or red-light violations. Still others have considered pedestrian safety, bicycle tracking, and helmet or seatbelt detection. These efforts demonstrate the versatility of deep learning in traffic applications.

However, a closer look at many of these works reveals that they either focus on a specific task (such as license plate reading or speed estimation) or expect relatively structured traffic. The unique characteristics of Indian mixed traffic high density, variable lane discipline, and frequent occlusions—remain under-represented in mainstream datasets and benchmarks. Additionally, many solutions stop at detection or counting, without offering integrated dashboards for daily operations or policy design.

Smart-city reports and pilot deployments indicate that city administrations are looking for systems that not only detect vehicles but also convert these detections into insights for planning. They expect tools that help answer practical questions, such as identifying junctions dominated by autos or segments where two-wheelers regularly spill over into oncoming lanes. *DeepTraff* aims to address this gap by combining a YOLO [1]–[6]-based detection pipeline with a flexible analytics dash- board, designed specifically with Indian mixed traffic in mind.

III. Related Work

While the previous section outlined the broader evolution of traffic monitoring solutions, this section provides a comparative examination of the most prominent categories of existing approaches, highlighting their operational techniques, application strengths, and shortcomings. Through this comparison, we clearly identify where *DeepTraff* fits within the current research landscape and emphasize its relevance to modern smart city deployments. Moreover, with the increasing adoption of smart city technologies in countries like India, the demand for automated, data-driven traffic insights has moved beyond mere vehicle detection to include real-time categorization, analytics, and decision support.

Traditional systems often generate fragmented or unstructured outputs, limiting their usefulness in managing congestion, tracking vehicle-type distribution, or planning infrastructure upgrades. In contrast, newer AI-based frameworks seek to convert raw video feeds into actionable information that can support planning, enforcement, and policy decisions. *DeepTraff* aligns with this emerging need by combining deep learning models with a structured analytics pipeline, making it not only a detection tool but a

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

comprehensive decision-support system for urban traffic management.

TABLE I
COMPARISON OF TRAFFIC MONITORING APPROACHES

Method	**Technique**	**Strengths**	**Limitations**
Traditional Sensors	Induction loops, radars	Reliable point measurements	High cost, limited coverage
Manual CCTV	Human observation	Context-aware judgments	Fatigue, non- scalable
Classical Vision	Background subtraction, optical flow	Lightweight, low-cost	Fails in cluttered scenes
Deep Detectors	Faster R-CNN [15]–[17], SSD	High accuracy	Higher compute load
YOLO-based	One-stage detection (YOLO [1]–[6])	Real-time, robust	Needs training, GPU
Dashboard	Detection + analytics	Decision support for planners	Requires integration ef- fort

DeepTraff falls squarely in the last category: it pairs a modern YOLO [1]–[6]-based detector with a dashboard and backend architecture focused on analytic usability and system integration.

IV. PROBLEM STATEMENT AND OBJECTIVES

In many Indian cities, CCTV cameras are already installed at key intersections under smart city programs. However, these devices are often used merely for recording, post-incident review, or at best, manual monitoring. They rarely produce structured, real-time data about traffic composition, category- wise flow, or congestion trends. In most surveillance rooms, operators manually watch multiple live streams, often without any analytical tool capable of converting footage to data that can support planning or operational decisions. As a result, hours of valuable video content remain underutilized, and insights that could guide road design, congestion management, or public transport policies remain inaccessible. This limited usage of existing infrastructure highlights the necessity for automating the transformation of video streams into useful traffic analytics.

The core problem addressed in this work is the absence of an integrated solution that not only detects vehicles in real time but also categorizes them into meaningful classes and presents interpretable analytics. Existing installations usually lack an end-to-end mechanism to transform raw video into actionable

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

information. In many cases, local authorities are dependent on manual counting surveys or expensive sensor- based deployments, even though cameras are already available. Such practices result in incomplete datasets, delayed decision- making, and operational inefficiencies.

The primary challenge addressed in this work is the lack of a unified framework that leverages existing CCTV infrastruc- ture to automatically detect, categorize, and convert real-time vehicle observations into meaningful traffic analytics suitable for operational decision-making.

- operates on existing CCTV infrastructure,
- performs real-time vehicle detection and categorization using YOLO [1]–[6],
- aggregates results into human-friendly analytics,
- supports heterogeneous Indian traffic conditions.

Based on this, the main objectives of *DeepTraff* are:

- to conceptualize an end-to-end pipeline from CCTV feed to dashboard insights;
- to employ deep learning for vehicle-type categorization (two-wheelers, three-wheelers, four-wheelers) using YOLO [1]–[6];
- to discuss dataset collection and annotation in an Indian city context;
- to outline deployment strategies across edge devices and cloud servers;

Furthermore, DeepTraff aims to bridge the gap between research-oriented detection systems and practical urban analytics tools. Instead of focusing solely on technical accuracy, the framework emphasizes operational usability, real-time visualization, and compatibility with policy-driven tools. The proposed system is envisioned not merely as a detector, but as a building block for data-driven governance and traffic management in Indian cities.

V. Proposed System

A. System Architecture

The proposed *DeepTraff* system is structured as a sequence of stages that transform raw video into action-oriented traffic insights. CCTV cameras continuously capture traffic scenes at one or more junctions. These cameras may be mounted on poles, traffic lights, or building corners, and stream video either through a wired network or an existing city surveillance backbone. The video feed is forwarded to a processing unit.

Within this processing unit, a YOLO-based deep learning detector executes inference on each frame. The model identifies vehicles, draws bounding boxes around them, and assigns category labels such as two-wheeler, three-wheeler, and four-wheeler. In addition to raw detection, the model also yields confidence scores that represent its certainty in each prediction. These detections are not only used for visualization but are also forwarded to an analytics layer running in the backend. The backend analytics layer aggregates detections across continuous frames to compute interpretable metrics such as vehicle density, class-wise percentage share, moving averages, and peak-hour trends.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

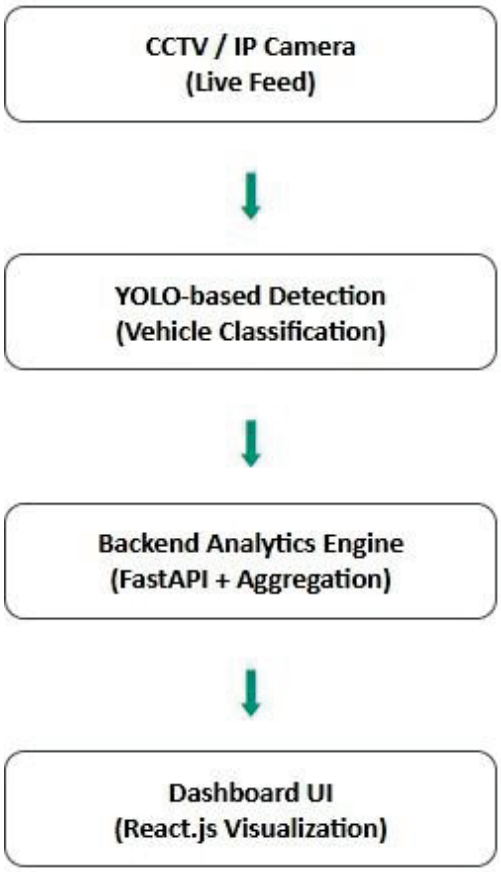

Fig. 1. Conceptual architecture of the DeepTraff system integrating CCTV feeds, YOLO [1]–[6]-based detection, backend analytics, and a dashboard.

It filters noise, corrects minor inconsistencies due to occlusions, and organizes statistics into small time windows suitable for operational use. These insights are then stored temporarily or passed directly to the frontend service.

Finally, a web-based dashboard allows city operators, traffic police, or planners to visualize this information in real time. The dashboard overlays detections on live footage, displays class-wise counters, showcases short-term trends through charts, and highlights congestion indicators. Because the system architecture follows a modular design, individual blocks (camera feed, detection model, backend analytics, or dashboard visualization) can be upgraded independently with- out altering the complete pipeline. This modularity enables *DeepTraff* to adapt easily to new camera locations, expanded vehicle categories, or integration with smart-city platforms.

B. Workflow

The overall workflow is depicted in Fig. 2. The main stages include:

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

1) **Video Acquisition:** Live feeds from roadside cameras.
2) **Frame Extraction:** Sampling frames at a suitable rate.
3) **Detection and Categorization:** Running a YOLO [1]–[6] model on each frame.
4) **Aggregation:** Combining detections into short time- window summaries.
5) **Visualization:** Displaying maps, charts, and live over- lays on the dashboard.

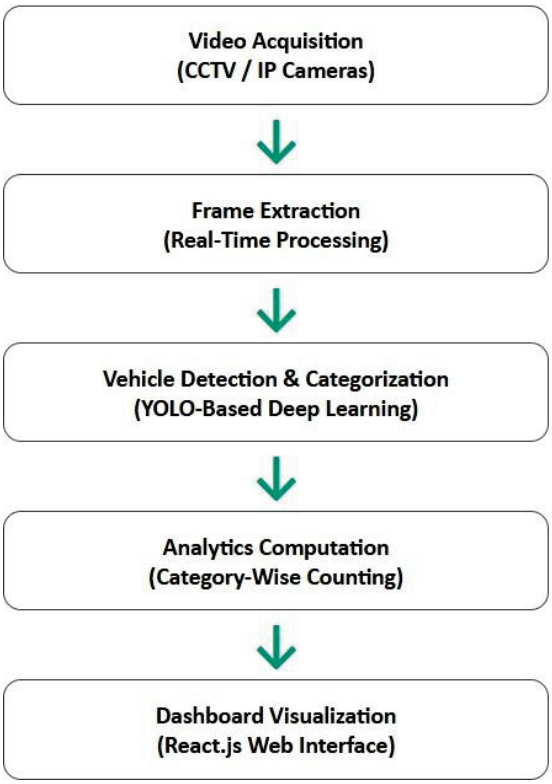

Fig. 2. End-to-end workflow followed by *DeepTraff* from camera feed to actionable analytics.

C. Dataset and Annotation

To ground the framework in realistic conditions, traffic footage was considered from a busy urban road segment in Moradabad, Uttar Pradesh, India. The chosen segment features typical Indian mixed traffic with motorcycles, scooters, public auto-rickshaws, private cars, and small vans. Video was captured across multiple time windows, including both peak and non-peak hours, to reflect natural variation in density and composition.

From this video, frames were sampled at regular intervals to avoid redundancy and to cover a diverse set of scenes. Ap- proximately 800–1000 frames were manually annotated using the LabelImg tool [27]. Each visible vehicle was enclosed in a bounding box and assigned to one of three categories:

- **Two-wheelers**: motorcycles, scooters, and similar vehicles;
- **Three-wheelers**: mainly public auto-rickshaws;

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

- **Four-wheelers**: private cars and compact vans.

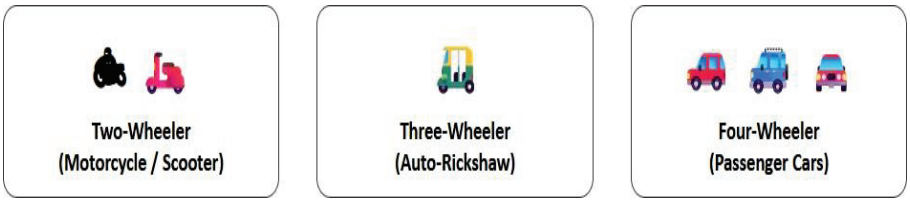

Fig. 3. Illustration of vehicle categories considered in *DeepTraff* (conceptual icons for two-wheeler, three-wheeler, and four-wheelers).

Care was taken to ensure that annotation captured a range of weather conditions, occlusion patterns, and traffic densities. Although the dataset is not exceptionally large, it is representative of the types of scenes that *DeepTraff* aims to handle. The dataset is conceptual in this paper and is not publicly released, ensuring that privacy concerns and local regulations are respected.

D. How DeepTraff Identifies Vehicles in Real Time

DeepTraff identifies vehicles by treating every video frame from the CCTV camera just like a visual scene that needs to be understood. Instead of relying on formulas or manual rules, the system directly learns how different vehicles look based on examples provided during training. The model observes features such as overall shape, wheels, colors, and size pat- terns that distinguish two-wheelers, three-wheelers, and four-wheelers commonly found on Indian roads.

When the system receives a new video frame, it immediately searches for familiar shapes. Once a vehicle is recognized, a colored box is drawn around it, and the category label is assigned automatically. Along with the label, the system also provides a confidence level that indicates how sure it is about its decision. These confidence values help ignore uncertain or misleading observations, especially in crowded road conditions.

This detection process runs repeatedly on every frame, allowing the system to continuously update the traffic scene. Instead of making decisions from a single frame, *DeepTraff* combines information from many frames occurring over a short period. This helps the system overcome temporary issues such as vehicles being blocked by others, sudden lighting changes, or fast movements. As a result, the output becomes more stable and reliable for real-world use.

The final outcome of this process is not just the identification of vehicles but a steady stream of categorized traffic information. These categorized detections are passed to the analytics layer, where they can be counted, compared, and visualized for planners and traffic police. Thus, without using any mathematical modeling, *DeepTraff* focuses on practical detection that works directly with live camera feeds, making it suitable for city-wide deployment without the need for specialized algorithm tuning.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

E. Backend and Dashboard Design

The backend of *DeepTraff* is conceptually implemented using a FastAPI-based service [25]. It receives detection outputs, stores them temporarily, and computes summary statistics such as per-category counts, moving averages, relative shares of each category, and simple congestion indicators. It also ex- poses REST endpoints [23], [24] through which the dashboard can request the latest analytics.

The dashboard, built using ReactJS [26], is responsible for presenting information in a clear and intuitive manner. Key features of the dashboard design include:

- a live video panel with bounding boxes and labels over- laid on the feed;
- counters showing current two-wheeler, three-wheeler, and four-wheeler counts;
- time-series charts depicting how these counts vary over the last few minutes or hours;
- pie or bar charts summarizing the percentage composition of traffic;
- panels to highlight peak periods and emerging congestion trends.

This design ensures that city operators do not need to interpret raw detections themselves; instead, they view aggregated, interpretable metrics that can support direct decision-making.

VI. Smart City Integration and Adaptive Signaling

DeepTraff is designed with smart city integration in mind. Modern urban governance platforms typically include modules for environmental monitoring, energy efficiency, public safety, and transportation. A well-integrated traffic analytics system can feed data into these platforms for more holistic planning. One important extension is the concept of adaptive traffic signaling. In a naive configuration, signal timings are fixed or vary only by time-of-day schedules. With category-wise analytics, a city can prioritize specific vehicle classes (for example, giving slightly more green time when buses or public transport vehicles dominate, or when large surges of two-wheelers are detected on certain approaches).

A virtual representation of the intersection receives continuous category-wise traffic estimates from *DeepTraff.* An optimization engine then proposes signal timing adjustments, which are communicated back to physical controllers. The same system can also suggest long-term changes like adding dedicated turning lanes or re-timing signals for school or market zones.

Additionally, *DeepTraff* analytics may be shared with other smart city services, such as:

- public transport scheduling systems,

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

- parking management platforms,
- emergency response routing,
- pollution monitoring and emission modeling.

This highlights that *DeepTraff* is not envisioned as a standalone tool, but as a component in an ecosystem of connected urban services.

VII. Deployment, Security, and Privacy

A. Deployment Scenarios

There are three primary deployment models for *DeepTraff*:

- **Edge-based Deployment:** The detector runs on a small GPU-enabled device such as an embedded AI board near the camera. Only aggregated statistics, and optionally low-resolution thumbnails, are sent to a central server. This reduces bandwidth and preserves privacy.
- **Cloud-based Deployment:** All video streams are sent to a centralized cloud or data center where a powerful GPU server processes multiple feeds simultaneously. This setup simplifies management but requires robust network infrastructure.
- **Hybrid Deployment:** Light pre-processing and detection happen at the edge, with deeper analytics and long-term storage handled in the cloud. This combines the strengths of both models.

B. Security and Privacy Considerations

Since *DeepTraff* analyzes CCTV footage from public roads, the system must be designed to protect the privacy of citizens and prevent unauthorized access to video data. Although the focus of the project is on vehicles and traffic patterns, cameras may still capture identifiable people, number plates, or other private details. Therefore, security and privacy become an essential part of the system rather than an optional feature.

To ensure responsible use, the system should incorporate multiple protection measures, including:

- encrypting video streams while they are being transmitted across networks, so that no external party can intercept or misuse the footage;
- restricting dashboard and backend access through strong authentication methods (login credentials, user roles, and permissions);
- anonymizing or blurring sensitive regions such as faces or number plates, especially when footage is

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

stored for long-term analytics or research;

- complying with national and local regulations related to digital surveillance, smart city monitoring, and responsible data usage.

By including these measures, *DeepTraff* not only delivers useful traffic insights but also ensures ethical deployment in real-world environments.

VIII. LIMITATIONS

While DeepTraff presents a promising framework, it is important to acknowledge its limitations:

- **Lighting and Weather:** Performance may degrade in heavy rain, dense fog, or very low light conditions, especially without infrared or thermal cameras.
- **Occlusions:** In extremely dense traffic, vehicles may be partially or fully occluded for many frames, making detection less reliable.
- **Hardware Requirements:** Real-time performance on multiple HD streams may require modern GPUs or specialized AI accelerators.
- **Dataset Scope:** The conceptual dataset described focuses on three broad categories; additional classes (buses, trucks, pedestrians, cyclists) would require more annotated data.
- **Domain Transfer:** A model trained on one city or camera angle may not immediately generalize to another location with different vehicle styles, camera heights, or backgrounds.

IX. FUTURE SCOPE

Numerous avenues exist for extending DeepTraff:

- **Additional Vehicle and Human Classes:** Incorporating buses, trucks, pedestrians, and cyclists to provide more complete road-user statistics.
- **Speed and Trajectory Analysis:** Using tracking to estimate velocities, detect harsh braking, and identify risky maneuvers.
- **Violation Detection:** Detecting red-light jumpers, wrong-lane driving, or helmet/seatbelt non-compliance.
- **Long-term Trend Analysis:** Archiving anonymized analytics to support seasonal, weekly, and event-based traffic studies.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

- **City-wide Deployment:** Scaling the system to many junctions and integrating with unified control centers.
- **Digital Twin Integration:** Building comprehensive simulation models that mirror real-world conditions using continuous *DeepTraff* inputs.

X. Conclusion

This paper has presented *DeepTraff*, a conceptual framework for real-time traffic sensing and vehicle category analytics tailored to heterogeneous Indian conditions. By combining YOLO [1]–[6]-based object detection with a modular back- end and an intuitive dashboard, *DeepTraff* demonstrates how existing CCTV infrastructure can be upgraded into a powerful decision-support system. The framework underscores the importance of analyzing the type and share of vehicles on the road instead of relying solely on coarse vehicle counts, which are often insufficient for modern transportation planning.

DeepTraff highlights the practical value of AI-based perception in cities where mixed traffic, frequent occlusions, and variable lane discipline pose challenges to traditional monitoring tools. The proposed design illustrates how real- time categorization of two-wheelers, three-wheelers, and four- wheelers can support critical activities such as congestion assessment, parking policy formulation, lane priority allocation, and public transport planning. Further, the system's modular workflow ensures adaptability with different deployment choices, whether edge-based, cloud-based, or hybrid, making it feasible for government agencies with varying budgets and infrastructure capacity.

Although the current work remains conceptual and does not provide quantitative performance evaluation, it lays a strong foundation for real-world implementation and academic research. Future studies may focus on dataset expansion, night- time monitoring, speed estimation, violation detection, multi- camera fusion, and integration with adaptive traffic signal control. As Indian cities migrate toward digital governance and smart mobility, solutions like *DeepTraff* can serve as a baseline model to catalyze data-driven decision making.

XI. References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. IEEE CVPR*, 2016.

[2] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *Proc. IEEE CVPR*, 2017.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 04, ICTEM 2.0 (2026)

[3] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *arXiv:1804.02767*, 2018.

[4] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv:2004.10934*, 2020.

[5] C.-Y. Wang et al., "YOLOv7: Trainable Bag-of-Freebies Sets New State- of-the-Art for Real-Time Object Detectors," *arXiv:2207.02696*, 2022.

[6] Ultralytics, "YOLOv8: Official Documentation and Model Zoo," GitHub, 2023.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE CVPR*, 2016.

[8] T.-Y. Lin et al., "Feature Pyramid Networks for Object Detection," in *Proc. IEEE CVPR*, 2017.

[9] T.-Y. Lin et al., "Focal Loss for Dense Object Detection," in *Proc. IEEE ICCV*, 2017.

[10] T.-Y. Lin et al., "Microsoft COCO: Common Objects in Context," in *Proc. ECCV*, 2014.

[11] A. Bewley et al., "SORT: Simple Online and Real-time Tracking," *Proc. IEEE ICIP*, 2016.

[12] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[13] Y. LeCun et al., "Gradient-Based Learning Applied to Document Recog- nition," *Proc. IEEE*, 1998.

[14] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proc. NIPS*, 2012.

[15] R. Girshick et al., "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *Proc. IEEE CVPR*, 2014.

[16] R. Girshick, "Fast R-CNN," in *Proc. IEEE ICCV*, 2015.

[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real- Time Object Detection with Region Proposal Networks," in *Proc. NIPS*, 2015.

[18] S. Gupta et al., "Deep Learning-Based Traffic Surveillance: A Review of Vision-Based Approaches," *Int. Journal of Computer Vision and Image Processing*, 2020.

[19] M. Choudhary et al., "AI-Driven Traffic Analytics for Smart City Monitoring," *Int. Journal of Smart Infrastructure*, 2023.

[20] R. Yadav et al., "Deep Learning Framework for Heterogeneous Indian Traffic Analysis," *Int. Journal of Information Technology*, 2022.

[21] A. Patil et al., "Edge-Based Intelligent Traffic Surveillance using YOLO and IoT," *IEEE Sensors Journal*, 2021.

[22] N. Kang et al., "Real-Time Automatic License Plate Recognition Using YOLO," in *Proc. IEEE ICSP*, 2017.

[23] R. T. Fielding and D. M. Taylor, "Principled Design of the Modern Web Architecture," in *Proc. Int. Conf. on Software Engineering*, 2000.

[24] F. Masse, *REST API Design Rulebook*. O'Reilly Media, 2011.

[25] FastAPI Documentation, "FastAPI: Modern Web Framework for Build- ing APIs with Python," 2024.

[26] Meta Platforms Inc., "React: A JavaScript Library for Building User Interfaces," 2024.

[27] LabelImg Project, "LabelImg: Open Source Image Annotation Tool," GitHub, 2015.

[28] Ministry of Housing and Urban Affairs, Govt. of India, "Smart Cities Mission: Strategy and Guidelines," Official Report.

[29] Various Authors, "Urban Mobility India Conference Proceedings," Govt. of India Publications.