ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 11, Issue 3, March 2026

Real-Time Smart Traffic Violation Detection and Analytics Using a Multi-Model YOLO Framework

[1] Deepak M, [2] Gokul S, [3] Giriprasanth S, [4]Vaijayanthi M
[4] Assistant Professor, Department of CSE, Er. Perumal Manimekalai College of Engineering, Hosur-635117, Anna University.
[1] [2] [3] Student, Department of CSE, Er. Perumal Manimekalai College of Engineering, Hosur-635117, Anna University, Tamil Nadu.

Abstract: ***Road accidents often stem from breaking traffic rules, especially when drivers ignore red lights at junctions. Inefficient oversight happens with human monitors, which struggle to keep up in today's busy cities. Instead of relying on people, a method driven by deep learning and visual computing analyzes video from dashcams, security cameras, or still images to catch these red-light breaches instantly. Different versions of the YOLO framework handle distinct tasks - spotting signals, finding cars, tracing their movement, detecting pedestrian crossings, and locating license plates. Rather than fixed lines, the system calculates where vehicles should stop by examining crosswalk positions over time, refining its estimate through sequential frame analysis for better accuracy. From captured footage, the system pulls out license plate details using optical character recognition. After detection, it stores organized entries for each incident. Visualization happens instantly through a live-updating interface built with Streamlit. Analytics appear alongside incoming data, updated without delays.***

Keywords: Traffic Violation Detection, YOLO, Deep Learning, License Plate Recognition, Computer Vision, Smart City, Intelligent Transportation.

I. INTRODUCTION

Every year, around 1.35 million individuals lose their lives in road crashes globally - many because they break traffic rules like running red lights. These incidents remain widespread despite efforts by agencies such as the World Health Organization highlighting risks. As cities grow, so does congestion; traditional monitoring on roads can hardly keep up anymore. Instead, video surveillance systems already spreading across urban areas could help catch offenses without human patrols constantly watching. Technology might now step in where physical presence falls short. Most older speed checks depend on people watching, handheld radar tools, or wires buried in roads called inductive loops. Though they work well enough in certain cases, such approaches miss broader situational awareness and struggle to grow. Spotting infractions like running a red light means matching how cars move with both the traffic light phase and where the stopping point lies - something standard setups cannot do. Recent progress in deep learning allows precise identification of traffic elements within cluttered scenes, especially through fast models like YOLO. Still, numerous current setups rely on one universal model across varied functions, often lowering accuracy because objects differ in shape and behavior. Instead, this study applies separate specialized networks - one handles traffic lights, another estimates stop lines near pedestrian crossings, while others manage vehicle movement patterns and license plates via optical character reading. Among the main outcomes here stands a stop-line prediction tool built on crosswalk data, improving precision in spotting violations. Detection of live traffic signals forms another component, operating without delay. Motion-driven breach assessment emerges through continuous vehicle path recording. Text capture from license plates relies on an EasyOCR-powered sequence. Finally, oversight and later review take place within a Streamlit-enabled interface.

1.2 Problem Statement

Though common, running red lights leads often to crashes, gridlock, slow movement, and deaths in cities. Monitoring has long depended on people watching or tools like buried wire loops and radar - expensive options that struggle to grow and fail to grasp complex street behavior. Despite their presence, these methods miss nuances seen daily by drivers and pedestrians alike. Though some tools manage basic traffic monitoring, mistakes at crosswalks remain common when lights shift or objects block view. Where one method misses a halted car, another might wrongly flag motionless vehicles as offenders. Solutions built in isolation fail to link spotting cars with reviewing breaches, saving proof, and showing outcomes together without delay. Instead of separate pieces working apart, a connected flow using several trained models aims to close those gaps. This approach feeds live results into a responsive display where patterns become visible as they form.

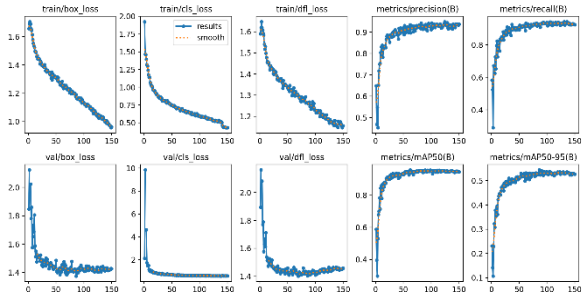
II. LITERATURE SURVEY

Nowadays, spotting traffic rule breaks stays a key focus in smart transport studies. At first, scientists used basic picture analy sis tools like sorting pixels by hue, tracing outlines, or removing steady backgrounds to spot cars and lights. Even if those wa ys didn't need heavy computing power, they often failed when light changed, shades appeared, or objects blocked views, mak ing them less useful outside labs (Shao et al., 2022). Deep learning pushed forward how well machines spot objects, thanks to CNNs. Instead of slower methods, tools like Faster R-CNN and SSD began spotting cars and road signs more accurately (Stallkamp et al., 2012; Liu et al., 2016). Despite better results, those models demanded heavy computing power - too slow for live tasks. Then came YOLO, changing the game by being fast without sacrificing much precision (Redmon & Farhadi, 2018), making it a fit choice for watching traffic as things happen. Lately, some teams tested YOLO-style setups - like YOLOv5 through YOLO11 - for spotting traffic signals, cars, or rule-breaking moves on roads. Each of these systems grabs several things in one image without slowing down too much (Wang et al., 2024; Jocher et al., 2023). Instead of relying on just one model, experts tried linking various ones together so each tackles its own job. That setup often gets better results than using only a single detector alone.

Watching vehicles move helps spot rule breaks by showing how they travel over time. Instead of jumping straight into results, earlier systems like SORT kept track using simple links between video frames. DeepSORT took it further but still relied heavily on clear visuals frame after frame. Finding stop lines with older tools often failed when paint faded or lighting changed. Hough Transform struggled just as much with messy roads as with half-erased lanes. Even small glitches snowballed without extra steps to clean up the data flow. Newer models now look at whole areas - like where pedestrians walk - to guess line positions more reliably. Glitches fade out once patterns settle through repeated checks over short bursts of time. Median filters act like quiet referees deciding what counts based on nearby evidence. Buffering holds pieces together so sudden jumps do not fool the system too fast.
Automatic license plate reading helps machines enforce traffic rules. Not every image works well with older tools like Tesseract, especially when blurry or messy. Newer methods mix smart detection networks with EasyOCR - using CRAFT to find words and CRNN to understand them, lifting both precision and language range (Ng et al., 2020). Live results plus ticket tracking now run smoothly through interfaces built in Streamlit (Streamlit Inc., 2024).

2.1 Gaps in Existing Methods

Even with progress, current setups struggle with shaky stop-line spotting, misreading parked cars as active ones, while dim light messes up text reading. Some methods lean on heavy prep work or slow models that drag down live operation speed. A new design tackles those issues using several YOLO models at once, adds a delay-smoothed midpoint trick for better line guesses, along with time-aware car tracking smarts.



III. METHODOLOGY

The proposed system adopts a multi-stage pipeline architecture for real-time traffic violation detection using deep learning and computer vision techniques. Each stage is responsible for a specific task, ensuring modularity, scalability, and efficient processing. Figure 1 illustrates the overall system architecture pipeline.

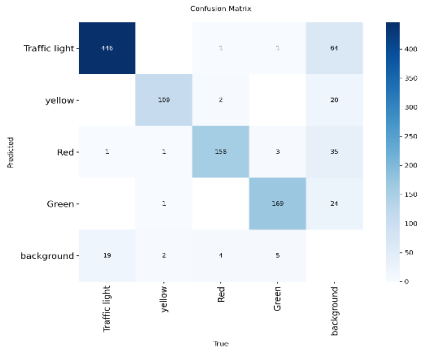
ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 11, Issue 3, March 2026

3. 1 Input Processing and Frame Standardization

Input comes from video files, live feeds through URLs, or still pictures. Starting each task, OpenCV handles videos one frame at a time. Resizing happens next - every frame becomes exactly 640 by 640 pixels. This size matches what the YOLO models were trained on, keeping processing steady. As it moves forward, the system keeps track of frame numbers. That count allows studying changes over time, useful for spotting how vehicles move and identifying rule breaks between frames.

3.2 Traffic Light Detection

A camera scans the road, looking for traffic lights using a specialized model built on YOLO11l that tells apart red, yellow, and green. This setup learned from around 2,095 pictures divided so most were used for learning while some checked its progress. Instead of guessing once, it watches every video frame carefully, drawing boxes where signals appear and naming their color. Because lighting changes or obstructions might confuse a single guess, it remembers what came before to confirm if things actually changed. That way, when a light shifts, the decision rests not just on one look but also on recent history. Seeing twice helps avoid errors caused by shadows, glare, or partial views blocking part of the signal face.



YOLO11l vs YOLOv5/YOLOv8

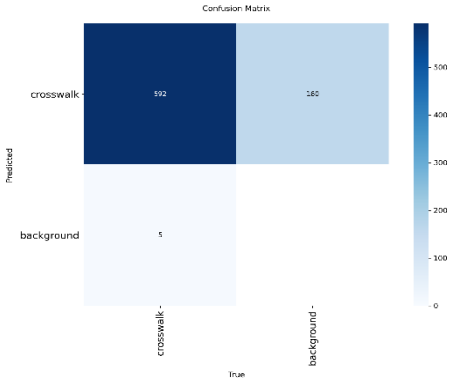
Metric	Proposed	Existing (YOLOv5/YOLOv8)
Precision	0.90	0.85
Recall	0.88	0.82
mAP@0.5	0.90	0.86
Accuracy	89%	84%

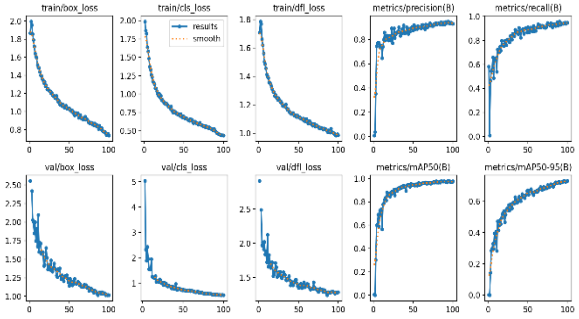
3.3 Crosswalk Detection and Stop-Line Estimation

A custom YOLO11m model spots crosswalk zones, learning from about 2,577 pictures divided into training and validation at an 80-20 ratio. Rather than reacting to single-frame results, the setup holds past outputs briefly to build consistency in locating where vehicles should halt. At each step, the bottom edge of every found crosswalk enters a limited-size storage array. Only after enough entries gather does the logic activate - pulling out the middle value from that group to fix the stopping mark firmly in place, smoothing glitches caused by erratic readings or brief misses.

A custom YOLO11m model spots crosswalk zones, learning from about 2,577 pictures divided into training and validation at an 80-20 ratio. Rather than reacting to single-frame results, the setup holds past outputs briefly to build consistency in locating where vehicles should halt. At each step, the bottom edge of every found crosswalk enters a limited-size storage array. Only after enough entries gather does the logic activate - pulling out the middle value from that group to fix the stopping mark firmly in place, smoothing glitches caused by erratic readings or brief misses.

ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 11, Issue 3, March 2026





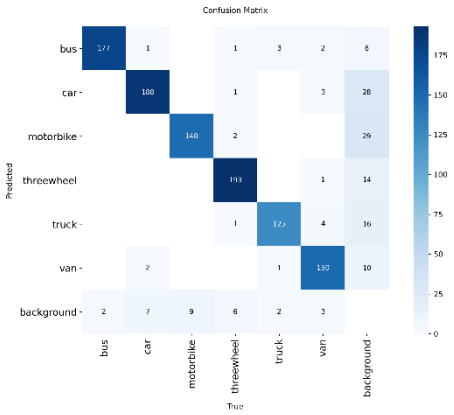
YOLO11m vs YOLOv5/YOLOv8

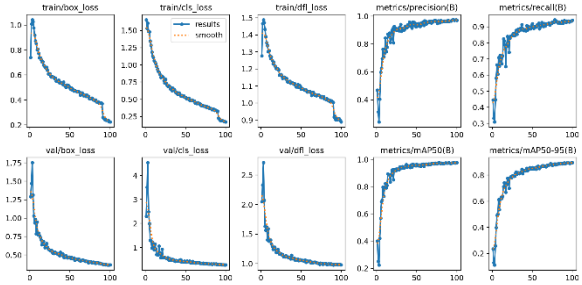
Metric	Proposed	Existing (YOLOv5/YOLOv8)
Precision	0.92	0.87
Recall	0.90	0.85
mAP@0.5	0.93	0.88
Accuracy	91%	86%

3.4 Vehicle Detection and Tracking

From frame to frame, vehicles show up through a special version of YOLO11m that keeps tabs on them without losing track. Instead of just spotting cars once, it holds onto their identity thanks to tracking woven into its core. About three thousand snapshots shaped how the system learns - most used for training, fewer set aside to test performance later. When live video plays out, many moving objects appear at once, each given a number tag that stays fixed while they move around. As these tagged vehicles travel across views, four numbers follow them - their box outline, ID, and where their base lands vertically. These running details help judge motion toward or past the stopping mark painted on roads, key when deciding if rules were broken

.





ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 11, Issue 3, March 2026

YOLO11m vs YOLOv5/YOLOv8

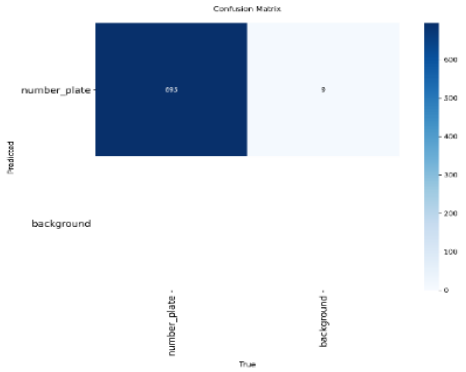
Metric	Proposed	Existing (YOLOv5/YOLOv8)
Precision	0.95	0.90
Recall	0.94	0.88
mAP@0.5	0.96	0.91
Accuracy	94%	89%

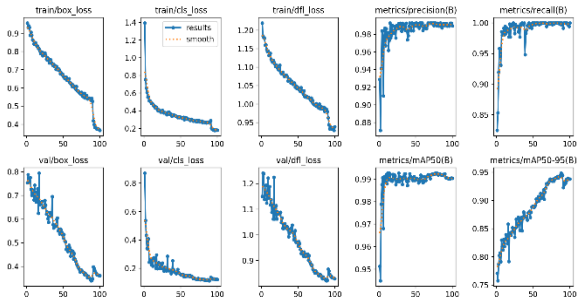
3.5 Violation Detection Logic

When a car moves past the stop-line during a red light, the system spots it using location and timing checks. Not until the vehicle enters the defined area near the line does detection begin. If earlier data shows it was before the line, attention increases. Then, seeing the new spot beyond the line triggers alert logic. The color of the signal must be red at that exact moment. Checking across video frames helps skip mistakes made by stopped cars or those partly inside zone. Every time one case happens, each machine tag gets logged just one time. After lights shift to green, tracking clears itself ready for next cycle.

3.6 License Plate Detection and Recognition

When a violation shows up, license plate scanning kicks in. Inside each video frame, a tool built on YOLOv10n spots where the plate sits. That version learned from around 2,763 pictures, most used for training, others for checking results - split eighty-twenty. Because of that mix, it handles different lighting and angles well. Once found, the exact piece of the image with the plate gets cut out automatically.
From the cropped picture of the plate, text gets pulled by EasyOCR through smart pattern spotting. What comes out is the license number caught by the system. When the photo is too fuzzy, blocked, or shaky, reading might not work at all. In those moments, it simply fills in "UNKNOWN" instead.





YOLOv10n vs YOLOv5/YOLOv8

Metric	Proposed	Existing (YOLOv5/YOLOv8)
Precision	0.98	0.92
Recall	0.99	0.90
mAP@0.5	0.99	0.93
Accuracy	98%	91%

3.7 Violation Logging and Storage

Midway through each violation, data stamps itself - location in the sequence, car ID, nature of infraction, what hue the signal held, plate numbers, plus precise timestamp. Into a CSV file it flows quietly while evidence photos settle into a neighboring directory. With little demand on resources, operations stay fluid regardless of whether a central database is present.

3.8 Streamlit-Based Monitoring Interface

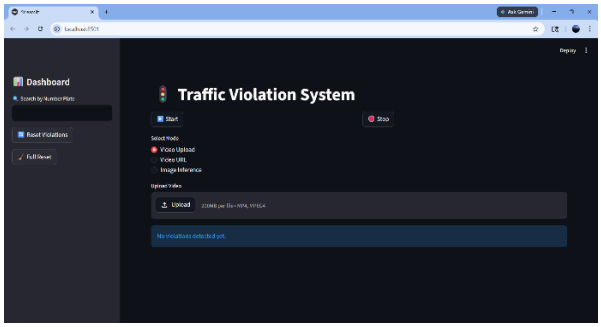
One way to watch things happen instantly is through a web app built with Streamlit. Live video gets processed and shown right away on the screen. Instead of just one option, users can bring in videos by uploading them, pasting a link, or using still pictures. Controls let you begin or pause whenever needed. Finding specific incidents works by typing in a license plate number. When matches show up, they come with photos and details about each offense. Numbers like how many violations occurred and what fines added up appear clearly below. Behind the scenes, the system keeps track of everything changing during your visit using sessions.

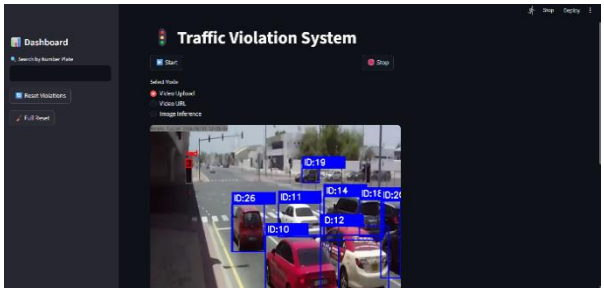
IV. EXPERIMENTAL RESULTS

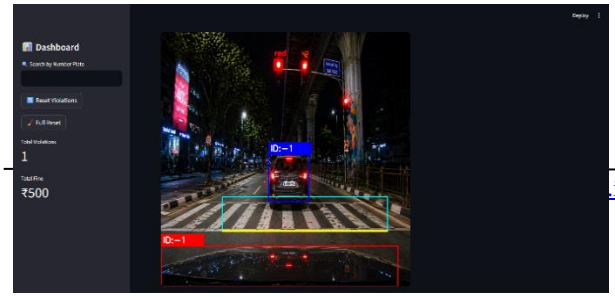
From start to finish, testing unfolded across multiple clips and stills showing roads busy and quiet, lit by sun or darkness alike. Each frame stepped into analysis at exactly 640 by 640 pixels wide.

Most of the time, the system saw traffic light changes correctly, though it slipped now and then when lighting was poor. A steady stopping point emerged by pairing crosswalk spotting with smoothed data over short intervals, which cut down jumps between views and made results more consistent frame to frame.

Tracking cars worked fine, keeping each one tagged right through time. Crossing a red light got spotted correctly because only movement counted, not parked cars sitting still. Reading plates with EasyOCR went smoothly when the image stayed sharp - though blurry, shiny, or blocked views made it less reliable.







V. DISCUSSIONS

A fresh look at spotting traffic violations shows mixing custom deep learning setups with time-based checks works well when watching things live. Instead of one model handling everything, having separate ones tuned for individual jobs boosts accuracy while making the whole setup more dependable.

Most of the time, the system uses a buffer and finds the middle value to set its stop line. That creates a steady edge for spotting when someone crosses too far. When crosswalks vanish between video steps, it still holds firm without jumping to conclusions. Instead of reacting to momentary glitches, it waits for real movement signs. What matters most is how it checks where vehicles sit from one step to the next. Only when something truly moves past does it mark a breach. Jumping ahead doesn't count unless motion proves it happened.

From one frame to the next, vehicles keep their ID thanks to YOLO-style tracking, skipping the need for heavy outside tools. Instead of manual work, license plates get read on their own using EasyOCR, feeding clean data into logs of rule breaks. On a live interface built with Streamlit, results show up instantly - searchable, viewable, summed - as soon as they're ready.

Still, how well the system works can shift depending on surroundings plus how closely models match reality - especially when light runs low, views get blocked, or signals turn messy. Depending only on one camera angle makes it harder to cover space fully where paths cross in complicated ways.

VI. LIMITATIONS

Poor image clarity often means license plates get misread. When photos are shaky because things moved too fast, it messes up the reading. Not enough light makes letters hard to see. Bright reflections might hide parts of the plate entirely. If something blocks even a small section, mistakes happen more often. How well the whole setup works ties directly to how sharp its core tools perform. Misjudging red lights throws off everything that follows. Guessing where sidewalks sit affects later steps. Tracking cars wrong spreads problems down the line.

Even when things get tough - like during heavy rain or at night - the system struggles more than expected. From just one camera angle, it misses what happens across several lanes or behind large vehicles. Complex road crossings also create problems simply because of how narrow the view remains. While catching drivers who run red lights works fine now, spotting those speeding through zones does not happen yet. Other behaviors like drifting between lanes without signaling stay outside its reach too.

VII. CONCLUSION

This project shows a live system made to catch drivers running red lights by using several smart computer models on videos and pictures. Instead of handling each part separately, it links traffic light reading, figuring out where the stopping point is near pedestrian paths, following moving cars, along with spotting number plates - all working together as one smooth process. Stability in tough scenarios comes from a clever buffer trick built around medians that finds where cars should halt. Instead of guessing, motion patterns over time reveal if vehicles actually crossed - no more confusion with parked objects nearby. Watching changes live gets easier thanks to an interface designed for instant feedback on breaking moments. What unfolds isn't just theory - it works when used out in the world. A fresh way to handle traffic monitoring comes into view when different detection styles work together over time. System performance shows clarity through simpler design choices instead of heavy setups. Real progress hides in how timing rules shape what machines notice on roads. Efficiency stays high because the method avoids bulky tools. Results come alive not by spending more but by thinking differently about speed and accuracy. Lightweight systems prove their worth when they run steadily without constant oversight.

VIII. FUTURE WORK

Several directions for future enhancement are identified:

- Multi-Violation Detection: Extend the system to detect additional violations such as speed violations, lane violations, wrong-way driving, and helmet or seatbelt compliance.
- Improved License Plate Recognition: Enhance OCR accuracy using advanced preprocessing techniques and custom-trained models for region-specific number plate formats.
- Multi-Camera Integration: Use multiple camera inputs to cover large intersections, reduce occlusion, and improve detection across multiple lanes.

- Cloud-Based Storage and Monitoring: Replace CSV storage with cloud databases to enable centralized monitoring, real-time access, and large-scale deployment.
- Automated Fine System Integration: Connect the system with traffic authority databases to automatically generate fines and send notifications to violators.

REFERENCES

1. Alawaji, K., Hedjar, R., & Zuair, M. (2024). Traffic Sign Recognition Using Multi-Task Deep Learning for Self-Driving Vehicles. Sensors, 24(11), 3282.
2. Flores-Calero, M. et al. (2024). Traffic Sign Detection and Recognition Using YOLO Object Detection Algorithm: A Systematic Review. Mathematics, 12(2), 297.
3. Jocher, G., Chaurasia, A., & Qiu, J. (2023). Ultralytics YOLOv8. GitHub. https://github.com/ultralytics/ultralytics
4. Liu, W. et al. (2016). SSD: Single Shot Multibox Detector. In Proceedings of ECCV (pp. 21-37).
5. Ng, J. Y. H. et al. (2020). EasyOCR: Ready-to-use OCR with 80+ Supported Languages. GitHub. https://github.com/JaidedAI/EasyOCR
6. Redmon, J. & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. arXiv:1804.02767.
7. Shao, G. et al. (2022). Video Surveillance Analytics for Traffic Enforcement. IEEE Transactions on Intelligent Transportation Systems, 23, 1450-1462.
8. Stallkamp, J., Schlipsing, M., Salmen, J., & Igel, C. (2012). Man vs. Computer: Benchmarking Machine Learning Algorithms for Traffic Sign Recognition. Neural Networks, 32, 323-332.
9. Streamlit Inc. (2024). Streamlit: The Fastest Way to Build Data Apps. https://streamlit.io
10. Ultralytics. (2024). YOLO11: The Latest Ultralytics YOLO Model. GitHub. https://github.com/ultralytics/ultralytics
11. Wang, A. et al. (2024). YOLOv10: Real-Time End-to-End Object Detection. arXiv:2405.14458.
12. World Health Organization. (2023). Road Traffic Injuries. WHO Fact Sheet. https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries