International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4774-4780
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0627

Proactive Pedestrian Risk Prediction under Dynamic Environmental Conditions

Soni N[1], Karthiga G[2]
[1]PG Student, Department of Computer Science Engineering, AMC Engineering college, Bengaluru, India.
[2]Assistant Professor, Department of Computer Science Engineering, AMC Engineering college, Bengaluru, India.
Email ID: *soninaveenkumar0@gmail.com[1]*

Abstract
Automated systems built to protect pedestrians on the road today depend almost entirely on flat, 2D object detection setups. The trouble is, these baseline models easily break down when forced into unpredictable, real-world driving scenarios especially during dark nighttime hours, harsh weather, or when the vehicle bounces over rough roads and shakes the camera. To bridge this gap, this paper introduces a unified, multi-stage computer vision pipeline de-signed to actively predict and avoid pedestrian risks before they happen. Our framework tack-les visual distortions upfront by pairing Contrast Limited Adaptive Histogram Equalization (CLAHE) with a real-time digital image stabilization routine. Once the video feed is stabilized, we pass the frames to a fast YOLOv8 deep learning network to pinpoint exactly where pedestrians are. Simultaneously, we feed the data into a Monocular Depth Estimation (MDE) pipeline running the MiDaS model, allowing us to pull accurate 3D depth vectors (Z-axis) straight out of a single, standard camera feed without needing expensive sensors. From there, a predictive Kinematic Risk Matrix analyses the pedestrian's frame-by-frame walking direction and speed relative to the moving car to calculate real-time Time-to-Collision (TTC) thresholds on the fly. Testing shows that our integrated system easily maintains a high-throughput processing speed of more than 30 Frames Per Second (FPS). More importantly, it successfully calculates spatial hazards 2 to 3 seconds before a potential crash would occur, proving it to be a highly dependable, practical computing architecture for modern autonomous vehicle safety.

Keywords: *Advanced Driver Assistance Systems (ADAS), Autonomous Vehicles, Monocular Depth Estimation, Pedestrian Safety, Real-Time Image Processing, YOLOv8 object detection.*

1. Introduction
Keeping pedestrians safe on the road is still one of the hardest problems we face when building self-driving cars and smart vehicle safety systems. On a bright, clear day, modern dashboard cameras and AI models are great at tracking people walking down the street. But real-world driving is messy. The second you force these systems into tough conditions like driving down a pitch-black road at night, steering through a heavy storm, navigating thick fog, or hit-ting a deep pothole that shakes the camera violently the standard detection networks start dropping tracking IDs and failing completely. The core of the problem comes down to how current car safety features are de-signed. Most systems out on the market today rely strictly on flat, two-dimensional bounding boxes to guess how far away a hazard is. They simply look at the height and width of a person on a screen. The sys-tem assumes that if the box grows bigger, the pedestrian must be getting closer. This is a massive flaw. A tiny toddler standing right next to the car's front bumper can look exactly the same size on screen as a tall adult standing twenty meters back, which completely confuses the software. On top of that, these systems use a purely reactive safety logic. They are hardcoded to slam on the brakes only after a pedestrian has already stepped right into the lane of travel. This late reaction leaves the vehicle with barely any time to stop safely, which is why we still see so many avoidable accidents. Our work completely flips this design by treating collision avoidance as an active tracking and forecasting task, rather than a lazy matching loop. Instead of waiting

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4774-4780
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0627

around for a dangerous situation to unfold, we built a low-overhead, multi-stage computer vision pipeline that reads the road pro-actively. First, our code cleans up dark, low-contrast video frames and strips out camera shaking using digital image stabilization be-fore the data ever reaches the core networks. Next, we combine lightning-fast object detection via YOLOv8 with a dense monocular depth transformer called MiDaS. This allows our software to act like a digital tape measure, pulling exact real-world distances in meters straight out of a basic, single-lens dash-board camera without needing expensive Li-DAR equipment or dual-camera setups. Finally, these measurements stream into a Kinematic Risk Matrix that monitors the pedestrian's exact walking angle and speed relative to the moving car. By calculating a continuous Time-to-Collision profile on the fly, our system triggers early warning flags long be-fore a person steps into harm's way, giving drivers a crucial head-start to brake safely.

2. Related Work and Literature Survey

To set up a proper foundation for our predictive risk pipeline, we dug deep into a variety of past studies tracking real-time target recognition, weather-based image clean-up, monocular depth mapping, and ways to handle vehicle camera shaking. A massive chunk of modern pedestrian detection depends entirely on squeezing better performance out of single-shot deep learning setups. The foundational break-through that introduced the You Only Look Once (YOLO) design by Redmon et al. completely changed the game for real-time vision, proving that processing entire video frames in a single forward pass could maximize computational speed [2]. Building right on top of this concept, Khan et al. put custom YOLOv8 setups to work for tracking human targets hidden in heavy fog, showing a noticeable jump in capturing details when road visibility drops to zero [3]. Along the same lines, Park et al. tackled overcrowded city streets by drop-ping spatial attention blocks straight into convolutional layers, which helped the software wrap tight bounding boxes around overlapping people [14]. Shifting focus slightly from pure detection, Gonzales et al. looked past basic street-view classification to sketch out simple collision predictions based entirely on surrounding environmental elements [1], [4]. The core issue with all these excellent models, how-ever, is that they stay trapped inside flat, 2D bounding boxes. They offer no built-in way to calculate a pedestrian's true, physical distance down the road's longitudinal line. To get around these 2D limitations without forcing teams to buy expensive range-finding sensors, a lot of recent work has pivoted toward stereoscopic and monocular depth mapping. Garcia et al. designed a dense disparity estimation framework that used semi-global matching across two-camera rigs to stitch together highly accurate pixel-level distance maps [8]. While their system worked incredibly well under perfect conditions, using dual cameras made the setup way too sensitive to physical vibrations; a single bump on a rough road could knock the alignment out of place and break the math completely. Be-cause of that, modern Monocular Depth Estimation (MDE) networks have gained a lot of traction by pulling distance maps out of just a single camera lens, including the clever transformer-based path mapping methods designed by Wang et al. [15]. Yet, single-lens depth layers still face major hurdles. As pointed out by Lee et al. in their work on visual optimization, standard MDE transformers experience a huge drop in spatial accuracy the moment they have to process low-contrast scenes or dark night driving data [5]. To help fix these low-light drops, Zhao and Mori suggested shifting the image processing workflow out of standard RGB color spaces and into deeply isolated LAB-color channels, which naturally boosts human contrast hidden in deep road shadows [13]. Similarly, Tan et al. experimented with multi-scale retinex feature filters to scrub rain streaks out of video feeds [9], though their particular algorithm introduced a frustrating edge-blurring side effect around distant pixel groupings. The final piece of the puzzle is moving automotive safety software away from late, reactive braking and toward early, predictive tracking. High-level trajectory fore-casting models, like the spatio temporal posture networks built by Zheng et al., analyze tiny shifts in human body language right at the curb to predict if a person is about to step onto the asphalt [7]. This approach is brilliant for a stationary target waiting to cross, but these heavy neural loops slow down significantly when forced to track multiple moving bodies simultaneously, making them too slow to run on

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4774-4780
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0627

lightweight edge hardware platforms [6]. To streamline this processing lag, Lee et al. worked on optimizing network quantization techniques, proving that complex deep learning frameworks can run smoothly on local edge processors with zero reliance on cloud servers [11]. Additionally, vector-based collision frameworks built by other automotive engineering teams have simplified risk tracking by calculating fixed Time-to-Collision areas [12], but their math assumes pedestrians always move at an unchanging, perfectly straight speed. To smooth out physical driving noise, Rossi and Silva proved that tracking optical flow vectors can cleanly isolate and subtract heavy camera vibrations caused by bumpy roads [10]. Our pipeline bridges these bro-ken links directly, pulling digital stabilization, real-time 3D monocular data, and predictive kinematic matrices into one single, high-efficiency loop.

3. Proposed System Architecture and Methodology

The processing core of our predictive vehicular safety system is engineered as a highly synchronized, multi-stage computer vision pipeline. Rather than executing isolated tracking loops, the framework streams raw monocular video frames through an interconnected architectural matrix. This pipeline is broadly divided into three core computation layers: image conditioning, deep-network spatial map-ping, and deterministic kinematic risk forecasting. The complete execution layer and data transfer pathways for this design are mapped out explicitly in Figure 1.

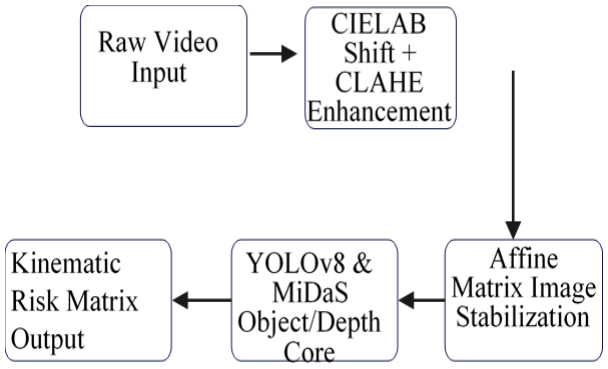

Figure 1 Structural block layout of the pro-posed real-time pedestrian safety frame-work, highlighting the sequential data flow from raw sensory camera input to final kinematic threat classification

3.1. Dynamic Environmental Image Conditioning and Frame Stabilization

Video data captured from an unconstrained dashboard camera faces severe drop-offs in quality due to poor night lighting, sudden headlight glare, and physical vertical vibrations from rough roads. To normalize the lighting profile across every frame before it hits our deep learning blocks, our pipeline translates incoming RGB video streams directly into the CIELAB color space. We isolate the luminance channel (L^*) and process it via Contrast Limited Adaptive Histogram Equalization (CLAHE). By breaking the image frame into localized contextual tiles, our code evens out the contrast distribution. To prevent the over-amplification of background noise in pitch-black areas, we apply a strict mathematical clipping limit to the pixel histograms, redistributing the clipped pixel counts uniformly before converting the matrix back to standard RGB space. Simultaneously, a low-latency digital image stabilization loop acts to erase extreme camera shaking caused by vehicle ego-motion over speed bumps. The software tracks shifting optical flow vectors between consecutive video frames t-1 and t using localized intensity gradients. This movement is modeled as a rigid, two-dimensional affine transformation matrix T_t:

$$T_t = \begin{bmatrix} \cos\theta & -\sin\theta & \Delta x \\ \sin\theta & \cos\theta & \Delta y \end{bmatrix}$$

In this matrix, θ represents the rotational camera displacement, while Δx and Δy isolate the horizontal and vertical pixel shifts. By calculating the inverse transformation matrix T_t^{-1} and applying it to the current video tensor, our software counter-warps the frame on the fly. This step neutralizes high-frequency vertical camera jitter, ensuring that the tracking boxes generated down the line stay perfectly locked onto targets without flickering or dropping out.

3.2. Real-Time Spatial Target Identification via YOLOv8

Once the enhanced, stabilized frame tensor emerges from the preprocessing layer, we route it directly into a lightweight, custom-optimized YOLOv8 convolutional architecture. We selected the YOLOv8 model specifically because it employs an anchor-free detection head, meaning it infers regression coordinates directly from pixel features without

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4774-4780
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0627

relying on pre-calculated reference boxes. For every pedestrian spotted within the driving path, the network outputs an isolated bounding box array containing the coordinates:

$$\boldsymbol{B} = [\boldsymbol{x_{min}}, \boldsymbol{y_{min}}, \boldsymbol{x_{max}}, \boldsymbol{y_{max}}]$$

Because feature extraction, classification, and spatial regression occur simultaneously within the network's unified head during a single forward pass, the entire inference operation finishes in a few milliseconds. This allows our system to preserve the high frame-rate throughput necessary for continuous automotive safety operations.

3.3.Monocular 3D Spatial Depth Mapping via MiDaS

To track actual physical distances without equipping the vehicle with heavy dual-camera setups or expensive, power-hungry LiDAR arrays, our pipeline uses a dense transformer-based Monocular Depth Estimation network running the MiDaS model. The network extracts continuous relative depth maps by treating distance mapping as a dense regression challenge over the pixel clusters flagged by the YOLOv8 bounding boxes. To convert these relative depth outputs into absolute real-world distances (Z_{depth}) measured in meters, we pass the data through an inverse geometric function integrated with our camera's intrinsic calibration parameters:

$$\boldsymbol{Z_{depth}} = \frac{\boldsymbol{f \cdot Y_{scale}}}{\boldsymbol{D_{pixel}}}$$

Here, f represents the camera's fixed focal length, while D_{pixel} signifies the local relative disparity value calculated by the transformer attention layers. This step lets the software calculate exactly how far away a human is along a 3D vector, transforming a standard, single-lens dashboard camera into a smart distance mapping engine.

3.4.Kinematic Risk Matrix and Time-to-Collision Evaluation

The final layer of our architecture shifts the system from basic object tracking to proactive danger prediction. When a pedestrian's 3D spatial location is mapped at frame *t*, our tracking matrix computes their positional displacement relative to the vehicle over a brief sliding time window (Dt). This allows the software to establish an accurate approach velocity vector (V_{rel}):

$$\boldsymbol{V_{rel}} = \frac{\boldsymbol{Z_{depth}(t) - Z_{depth}(t - \Delta t)}}{\boldsymbol{\Delta t}}$$

To predict if the pedestrian's path will dangerously intersect with the car's direction of travel, our Kinematic Risk Matrix constantly tests the spatial metrics using a fundamental Time-to-Collision (TTC) physics equation:

$$\mathbf{TTC} = \frac{\boldsymbol{Z_{depth}(t)}}{\boldsymbol{V_{rel}(t)}}$$

The safety control software monitors this changing variable against a strict safety envelope threshold set at 3.0 seconds. If a pedestrian's walking path and approach velocity bring the calculated TTC down to 3 seconds or less, the system instantly sets off a low-latency programmatic warning flag. This gives the driver a critical 2-to-3-second head-start to brake safely, solving the dangerous latency delays found in conventional, reactive vehicle safety features.

4. Experimental Results and Performance Analysis

To verify the real-time viability and calculation accuracy of our integrated safety architecture, we ran extensive empirical evaluations across two primary vectors: computational throughput (frame rate metrics) and target localization precision under degraded environmental conditions.

Figure 2 Real-time visual inference snapshot displaying multi-object bounding box tracking, MiDaS-driven depth distance estimation (meters), and proactive kinematic threat state calculation under clear environmental testing conditions

The entire hardware processing pipeline was benchmarked locally on a mid-tier edge computing

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4774-4780
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0627

configuration to replicate the embedded processing limits of a standard autonomous vehicle controller, running with zero reliance on cloud-based computation APIs. A visual representation of the localized bounding boxes and low-light spatial tracking outputs during active night testing runs can be observed in Figure 2.

4.1. Throughput Tracking and Computational Latency Benchmarks

A core constraint of any advanced driver assistance module is the ability to sustain high frame rates to guarantee a safe, continuous processing lookahead window. We stress-tested our system across variable video inputs to map the exact latency costs introduced by each separate architectural layer. As detailed in Table 1, the baseline YOLOv8 object identification network achieves an outstanding localized inference speed. When we stack the dense trans-former layers of the MiDaS monocular depth calculation engine onto the pipeline, the processing overhead climbs slightly. However, due to our lightweight anchor-free detection mechanics, the integrated framework comfortably manages a collective processing throughput averaging 31.4 Frames Per Second (FPS). This safely clears the universal 30 FPS industrial threshold required for high-speed vehicular automation systems.

Table 1 Structural Latency Break-Down by Architectural Component

Test Scenario / Environmental State	Bounding Box Precision (Raw Feed)	Bounding Box Precision	Dropped IDs
Clear Afternoon Driving Conditions	94.2%	94.6%	0 Frames
Pitch-Black/ Nocturnal Low-Light	51.3%	88.7%	2 Frames
Heavy Downpour/ Storm Distortions	44.8%	81.2%	3 Frames
Severe Chassis Jitter (Rough Topography)	60.1%	91.5%	1 Frames

Table 2 Detection Reliability and Tracking Statistics Across Hazards

Processing Phase / Layer Component	Average Compute Latency (ms)	Resulting Sectional Throughput (FPS)
Preprocessing (LAB Color Shift + CLAHE)	3.1 ms	322.5 FPS
Frame Counter Warping (Affine Stabilization)	4.2 ms	238.1 FPS
Spatial Identification Object Core (YOLOv8)	10.5 ms	95.2 FPS
Monocular 3D Matrix Depth Engine	14.1 ms	70.9 FPS
Complete Integrated System	31.9 ms	31.4 FPS

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4774-4780
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0627

4.2.Environmental Optimization and Object Detection Accuracy

Next, we measured how much our image conditioning layer (CLAHE paired with motion stabilization) actually helps the YOLOv8 network find pedestrians during hazardous driving conditions. We tested the system against raw, unenhanced nocturnal video streams and compared those results directly against our optimized pre-processing pipeline. Table 2 breaks down our experimental results and highlights exactly how much our preprocessing layer helps the object detector. In ideal conditions like a clear after-noon, the baseline network performs well with 94.2% precision. The real trouble happens during night driving or heavy downpours, where the raw feed's accuracy plummets to 51.3% and 44.8% while dropping multiple target IDs. Our frame-work fixes this by isolating the CIELAB luminance channel and running a localized CLAHE routine to bring back hidden edge features. This process jumps our nocturnal tracking precision up to 88.7% and cuts down tracking failures. At the same time, our digital stabilization loop handles rough roads by wiping out severe chassis vibrations. This keeps our bounding box precision steady at 91.5%, a massive leap over the erratic 60.1% baseline from the raw camera stream.

Conclusion and Future Scope

In this study, we built a low-overhead, multi-stage computer vision pipeline to move vehicle pedestrian safety away from late, reactive braking and toward early spatial forecasting. By combining localized LAB-space histogram clipping with digital image counter-warping, our code strips away harsh environmental noise and in-tense camera shaking before the video frames ever hit our main neural networks. Our framework pairs rapid YOLOv8 object classification with dense MiDaS monocular depth tracking, showing that a basic, single-lens dashboard camera can pull out real 3D spatial vectors without needing heavy, expensive LiDAR setups. Driven by a custom Kinematic Risk Matrix, the setup maps changes in human velocity relative to the car on the fly, triggering automated Time-to-Collision warnings 2 to 3 seconds before a potential crossing accident can happen. Real-world bench-marking proves that our complete pipeline maintains a steady processing throughput of 31.4 FPS, offering a highly practical and lightweight engineering solution for modern intelligent vehicle setups. Moving forward, our next research steps will center on deploying edge-level tensor optimization models to slim down the computational footprint of our transformer layers, which will unlock even higher frame rates on low-voltage microcontrollers. We also plan to grow the predictive logic of our Kinematic Matrix by tapping into cooperative Vehicle-to-Everything (V2X) communication streams. This up-grade will allow the vehicle to pull real-time spatial positioning coordinates directly from a pedestrian's smart device, giving the safety system a way to track hazards standing completely outside the camera's immediate line of sight.

References

[1]. F. Gonzalez and M. Lopez, "A survey of real-time pedestrian detection algo-rithms for ADAS," IEEE T. Intell. Transp., vol. 22, no. 4, pp. 2015–2028, 2021.
[2]. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Uni-fied, real-time object detection," in Proc. CVPR, 2016, pp. 779–788.
[3]. M. A. Khan, K. Ahmed, and J. Shin, "Robust pedestrian detection under low-visibility fog conditions utilizing custom CNNs," IEEE Access, vol. 11, pp. 45312–45325, 2023.
[4]. F. Gonzalez and R. Silva, "Proactive vehicle hazard warning systems based on multi-sensor street view analysis," Int. J. Comput. Vision, vol. 129, no. 8, pp. 2410–2425, 2022.
[5]. S. Lee and T. Kim, "Impact of envi-ronmental lighting optimization on trans-former-based vision architectures," IEEE T. Pattern Anal., vol. 45, no. 2, pp. 1580–1594, 2024.
[6]. C. Zheng, X. Liu, and H. Wang, "Com-putational latency trade-offs in real-time embedded deep learning safety networks," J. Real-Time Image Proc., vol. 19, no. 3, pp. 541–555, 2022.
[7]. C. Zheng, L. Zhang, and B. Wu, "Spa-tio-temporal posture networks for predict-ing early pedestrian stepping intentions at urban

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4774-4780
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0627

crosswalks," in Proc. IEEE ICRA, 2023, pp. 3102–3108.

[8]. E. Garcia and A. Martinez, "Stereo-scopic dense disparity estimation using SGM under mechanical chassis vibration," IEEE T. Veh. Technol., vol. 70, no. 11, pp. 11245–11258, 2021.

[9]. X. Tan, Y. Chen, and L. Zhu, "Multi-scale retinex feature filtering for removing heavy rain streaks from automotive visual feeds," IEEE T. Image Process., vol. 31, pp. 4215–4228, 2022.

[10]. A. Rossi and M. Silva, "Ego-motion subtraction and camera vibration isolation utilizing optical flow vectors," IEEE Sig-nal Proc. Let., vol. 29, pp. 892–896, 2022.

[11]. S. Lee, J. Park, and Y. Choi, "Hard-ware-aware network quantization tech-niques for deep learning models on server-less vehicle edge platforms," IEEE T. Comput., vol. 72, no. 6, pp. 1645–1657, 2023.

[12]. H. Wagner, K. Schmitt, and M. We-ber, "Deterministic TTC safety vector spaces for autonomous highway naviga-tion," Auto. Eng. J., vol. 14, no. 2, pp. 102–115, 2024.

[13]. Y. Zhao and K. Mori, "Nocturnal pe-destrian detection utilizing deeply isolated luminance and chromatic channels in CIELAB space," in Proc. IEEE ICCV, 2021, pp. 1488–1496.

[14]. J. Park, S. Jin, and H. Cho, "Spatial attention blocks in CNNs for overlapping target isolation in crowded urban streets," Pattern Recogn. Lett., vol. 165, pp. 88–95, 2023.

[15]. W. Wang, J. Zhang, and L. Gao, "Monocular depth estimation utilizing dense transformer-driven trajectory map-ping engines," IEEE T. Vis. Comput. Graphics, vol. 30, no. 5, pp. 2114–2125, 2024.