ISSN 2319-5991 www.ijerst.com

Peer Reviewed, Referred & Indexed Journal

Research Paper

AI-Powered Image Dehazing and Vehicle Detection in Foggy Road Conditions

[1] GUNDETI SWAMY, [2] K. VARUN REDDY,

[3] K. SISINDHRI, [4] L.SRI RAM

GUDIE: Mr. M. SAI KRISHNA (Assistant Professor)

Department of Electronics and Communication Engineering,

TKR College of Engineering and Technology, Medbowli, Meerpet, Balapur, Hyderabad, Telangana, 500097, India

ABSTRACT

Road safety is often affected by weather conditions such as smog, fog, and smoke, which reduce visibility and make driving risky. Poor visibility increases the chances of accidents, slows down traffic, and creates difficulties for emergency services. In foggy or smoky conditions, drivers struggle to detect vehicles, pedestrians, or obstacles in time. Traditional aids like fog lights and reflectors offer only limited support, leaving a gap for more effective, technology-driven solutions that can work in real time to improve road safety.

This project proposes a system that performs real-time haze removal and object detection. The method combines image enhancement with artificial intelligence to provide clearer visuals and better detection accuracy. First, a dehazing algorithm is applied to remove the effect of smog or fog and restore hidden details in the video frames. Once the visibility improves, the YOLO (You Only Look Once) object detection model identifies vehicles, pedestrians, and other objects with greater accuracy. This two-step process ensures that the system works effectively even in low-visibility conditions, providing reliable results for road monitoring and safety.

Index Terms- Image Dehazing, Dark Channel Prior, Atmospheric Scattering Model, Guided Filter, YOLOv8, Object Detection, Foggy Road Vision, Transmission Map, Real-Time Processing, Embedded Vision Systems, Intelligent Transportation.

I. INTRODUCTION

The safety of road transport systems relies critically on the ability of both human drivers and automated sensing platforms to perceive the surrounding environment with sufficient clarity and speed. Atmospheric haze encompasses a range of aerosol-laden conditions including fog, mist, smog, and dust storms that suspend light-scattering particles in the lower atmosphere. The physical consequence of aerosol suspension is a degradation of captured imagery characterised by reduced luminance contrast, colour desaturation, and progressive blurring of object boundaries with increasing scene depth.

These adverse visual conditions directly affect road safety by increasing driver reaction times and impeding hazard recognition. In countries such as India, regions like the Indo-Gangetic Plain routinely experience dense winter fog that renders highway sections dangerous and requires temporary speed restrictions and convoy protocols. Beyond human drivers, the expanding ecosystem of camera-based Advanced Driver Assistance Systems (ADAS), traffic surveillance networks, and prototype autonomous vehicles introduces a class of electronic observers whose performance is acutely sensitive to image degradation. Deep learning-based detection models are trained largely on clear-weather imagery, and their accuracy deteriorates markedly when the input distribution shifts to include haze-corrupted frames.

A principled strategy for addressing this degradation is to interpose a computational restoration stage before the detection network, recovering a haze-free approximation of the scene from the degraded observation. This preprocessing approach, broadly termed image dehazing or defogging, allows a detection model trained on clear imagery to operate within its intended statistical regime. Among the available dehazing techniques, physics-based methods such as the Dark Channel Prior offer the advantage of requiring no training data or GPU-accelerated inference, making them particularly attractive for embedded or resource-constrained deployment contexts.

This paper presents an integrated dehazing and detection pipeline designed for real-time vehicle detection in foggy road video. The system pairs the Dark Channel Prior dehazing algorithm, augmented with guided filter-based transmission map refinement, with a YOLOv8 nano object detection model. The contributions are: (1) a complete, CPU-deployable Python implementation of the DCP pipeline from first principles; (2) integration with class-selective YOLOv8 detection in a streaming video loop; (3) empirical evaluation of image quality and detection performance gains attributable to the dehazing stage; and (4) feasibility analysis for embedded deployment on the Raspberry Pi 5 platform.

II. BACKGROUND AND RELATED WORK

2.1 Atmospheric Scattering Model

The physical foundation for all physics-based dehazing algorithms is the atmospheric scattering model, which describes haze image formation as a function of depth-dependent light attenuation and ambient scatter. When scene radiance travels from a surface at depth d toward a camera, suspended aerosol particles scatter a portion of that radiance out of the optical path, attenuating it exponentially according to Beer-Lambert's Law. Simultaneously, ambient illumination scattered into the camera's field of view adds an additive airlight component that elevates apparent pixel intensity uniformly, suppressing local contrast. The standard formulation relating the observed hazy image I(x) to the latent haze-free scene radiance J(x) is:

$$I(x) = J(x) \cdot t(x) + A \cdot (1 - t(x))$$

where t(x) is the per-pixel medium transmission coefficient ranging from 0 (opaque haze) to 1 (clear air), and A is the global atmospheric light vector encoding the colour and intensity of ambient scatter. Recovering J(x) from I(x) requires simultaneous estimation of A and t(x) from the degraded observation alone.

2.2 Dark Channel Prior

He, Sun, and Tang demonstrated that in the vast majority of local patches drawn from haze-free outdoor imagery, at least one colour channel at one or more pixels attains a value very close to zero. This statistical regularity, termed the Dark Channel Prior (DCP), arises from the physical presence of shadows, dark foliage, colourful surfaces, and architectural elements that strongly absorb in at least one spectral band. In hazy images, the additive airlight term systematically elevates all channel values, raising the dark channel in proportion to the local haze density and in inverse proportion to the local transmission. By exploiting this relationship, the dark channel of the hazy image normalised by the atmospheric light estimate provides a direct analytical route to the transmission map. A guided filter post-processing step then refines the raw transmission map to preserve edge structure at depth discontinuities while suppressing patch-based block artefacts.

2.3 Learning-Based Dehazing

Data-driven approaches, exemplified by DehazeNet and the All-in-One Dehazing Network (AOD-Net), learn direct mappings from hazy to clear images using convolutional neural networks trained on paired synthetic datasets. AOD-Net reformulated the atmospheric scattering equation to yield a single network that jointly estimates atmospheric light and transmission in one forward pass. While learning-based methods achieve superior benchmark performance, they require GPU hardware for real-time operation, depend on representative training data, and are less interpretable than their physics-based counterparts. For embedded deployment scenarios with no GPU and no access to labelled foggy training data, physics-based dehazing provides a more practical alternative.

2.4 Object Detection in Adverse Weather

Standard object detection benchmarks such as MS COCO and Pascal VOC consist predominantly of clear-weather imagery, providing little exposure to the degraded inputs encountered in foggy or hazy conditions. Research on domain-adapted detection for fog, including the Fog-YOLO architecture, has shown that standard YOLO models trained on clear imagery suffer pronounced precision and recall reductions when applied directly to foggy scenes, with the performance gap most severe for small and distant objects. The dehazing-then-detection pipeline proposed in this work addresses this distribution mismatch by restoring the input to a haze-free approximation before detection, enabling a standard high-performing model to operate within its trained input regime.

III. SYSTEM DESIGN AND METHODOLOGY

3.1 Pipeline Architecture

The proposed system follows a sequential modular pipeline comprising five stages: (1) video frame acquisition via OpenCV VideoCapture; (2) dark channel computation and atmospheric parameter estimation; (3) transmission map construction and guided filter refinement; (4) haze removal and frame reconstruction; and (5) class-selective vehicle detection and annotation. Each module receives well-defined inputs and produces clearly specified outputs, permitting independent testing and replacement of individual components.

Frames are downscaled by a configurable factor prior to the computationally intensive dehazing stages, since atmospheric parameters vary smoothly across the image and can be estimated accurately at reduced resolution. The transmission map is subsequently upscaled to the original frame dimensions before the guided filter refinement step, which requires full-resolution guidance from the grayscale hazy frame to preserve object-boundary sharpness. The dehazed full-resolution frame is then submitted to the detection model.

3.2 Dark Channel Computation

The dark channel of an image patch is defined as the minimum intensity value across all three colour channels within a local neighbourhood. In the implementation, the DarkChannel function separates the BGR image into its three channels, computes a pixel-wise channel minimum using element-wise operations, and applies morphological erosion with a square structuring element of configurable patch size. Erosion is mathematically equivalent to a spatial minimum filter and is executed efficiently through OpenCV's optimised C++ backend.

3.3 Atmospheric Light Estimation

Accurate estimation of the global atmospheric light vector A is critical to the correctness of the transmission map and subsequent image recovery. The procedure identifies the top 0.1% of pixels ranked by dark channel value; these pixels are most likely to lie within regions of dense haze rather than on bright object surfaces. Among these candidates, the pixel with the highest total intensity in the original hazy image is selected as the atmospheric light estimate. This selection criterion is robust to non-haze bright regions such as specular highlights, because the dark channel constraint restricts consideration to genuinely hazy locations. The result is a three-element BGR vector representing the spectral composition of ambient scatter, typically close to uniform grey with a slight blue bias.

3.4 Transmission Map and Guided Filter Refinement

Given the atmospheric light estimate A, the transmission map is constructed by normalising the hazy image by A and computing the dark channel of the normalised image, combined with an omega weighting factor (set to 0.95) that retains a small residual haze at large depths to preserve perceptual depth cues. The resulting raw map exhibits block artefacts from the patch-wise minimum operation and does not accurately follow depth boundaries at object edges.

To refine the map, a guided filter is applied using the grayscale luminance of the full-resolution hazy frame as the guidance signal. The guided filter models the filtered output as a local linear function of the guidance image, solved via a regularised least-squares formulation in each local window of radius r. The regularisation parameter epsilon controls the sharpness of edge transfer; smaller values produce sharper depth boundaries in the refined map. The implementation computes local statistics using OpenCV box filters, yielding a refined transmission map that is spatially smooth in homogeneous regions and edge-preserving at object boundaries.

3.5 Image Recovery

Scene radiance is recovered by inverting the atmospheric scattering equation:

$$(x), t_0) + A$$

where $t_0 = 0.1$ is a minimum transmission floor preventing numerical amplification of noise in regions where the transmission approaches zero. The operation is applied in a fully vectorised manner using NumPy broadcasting, and the result is clipped to [0, 255] and cast to uint8 for compatibility with downstream processing.

3.6 YOLOv8 Nano Detection

YOLOv8, developed by Ultralytics, is the eighth major iteration of the You Only Look Once single-stage detection family. The nano variant (yolov8n.pt) employed in this work contains approximately 3.2 million parameters, providing a favourable balance between detection capability and inference latency on CPU hardware. The architecture employs an anchor-free detection head that directly predicts bounding box centres, widths, and heights from grid cell positions, eliminating dependence on manually defined anchor priors. A modified CSPDarknet backbone with C2f blocks performs multi-scale feature extraction, and a path aggregation network neck fuses features across scales to support detection at three strides (8, 16, and 32 pixels) covering small, medium, and large object sizes. The model is pre-trained on the 80-class MS COCO dataset and applied here with class-selective filtering restricted to COCO identifiers 2 (car), 3 (motorcycle), 5 (bus), and 7 (truck).

IV. IMPLEMENTATION

4.1 Development Environment

The system was developed and tested using Python 3.10 on a Windows 10 workstation. The core library dependencies are summarised in Table I. No GPU hardware is required for the physics-based dehazing components, which rely entirely on NumPy vectorised operations and OpenCV's C++ implementations. YOLOv8 inference is performed on CPU using the PyTorch backend provided by the Ultralytics package.

TABLE I: Software Dependencies

Library	Version	Role
OpenCV (cv2)	4.8+	Frame I/O, morphological ops, display
NumPy	1.24+	Vectorised dehazing computations

Ultralytics	8.0+	YOLOv8 model loading and inference
PyTorch	2.0+	Neural network inference backend

4.2 Key Implementation Functions

The DarkChannel function splits the input floating-point BGR image into three channels, computes a pixel-wise minimum across channels, and applies morphological erosion to produce the spatial minimum within each local patch. The AtmosphericLight function uses NumPy argpartition for O(n) identification of the top dark-channel candidates, avoiding the O(n log n) cost of a full sort. The TransmissionMap function normalises the image by the atmospheric light vector and applies the dark channel to derive the initial transmission estimate. The ManualGuidedFilter function implements the guided filter using five OpenCV box filter calls to compute the local statistics required for coefficient estimation. Finally, the Dehaze function applies the recovery equation in a fully vectorised manner with a minimum transmission clamp of 0.1 to prevent noise amplification.

The main processing loop instantiates a single YOLO model before the loop begins to avoid repeated weight loading. Streaming inference mode (stream=True) is activated to return a generator of result objects rather than a list, reducing peak memory consumption for long video sequences. The per-frame output concatenates the original hazy frame and the annotated dehazed frame side-by-side for operator comparison.

4.3 Parameter Configuration

TABLE II: Pipeline Parameter Settings

Parameter	Value	Effect
window_size	15 px	Patch size for dark channel spatial minimum
omega	0.95	Residual haze retention factor at depth
scale	0.50	Resolution reduction for DCP estimation stage
radius	40 px	Guided filter spatial smoothing extent
eps	0.001	Guided filter regularisation; controls edge sharpness
t_0	0.10	Minimum transmission floor for numerical stability

V. HARDWARE DEPLOYMENT CONSIDERATIONS

5.1 Target Platform: Raspberry Pi 5

The Raspberry Pi 5 serves as the target embedded deployment platform, featuring a quad-core Arm Cortex-A76 processor at 2.4 GHz, 8 GB LPDDR4X SDRAM, dual MIPI CSI-2 camera interfaces, and USB 3.0 connectivity. The absence of a dedicated GPU requires all processing to execute on the CPU cores, making computational efficiency a primary design constraint.

Benchmarking on the Raspberry Pi 5 indicates that DCP dehazing on a 1280×720 frame at 0.5× scale requires approximately 120–180 ms per frame. YOLOv8 nano inference contributes an additional 400–600 ms per frame in CPU-only mode, yielding a combined pipeline latency of 500–780 ms and an effective frame rate of approximately 7–8 fps. This rate is insufficient for smooth real-time video at 25–30 fps but adequate for snapshot-based traffic monitoring applications operating on infrequent frame sampling.

V. RESULTS AND ANALYSIS

5.1 Dehazing Quality Metrics

Quantitative evaluation of dehazing quality was performed using frames from the RESIDE benchmark dataset for which ground-truth haze-free reference images are available. Three metrics were computed: PSNR, SSIM, and the Colour Naturalness Index (CNI). PSNR measures the logarithmic ratio of maximum signal power to mean squared reconstruction error. SSIM jointly assesses luminance, contrast, and structural fidelity between the dehazed output and the reference image. CNI is a no-reference metric quantifying the naturalness of the output colour distribution. The results are summarised in Table III.

TABLE III: Dehazing Quality Metrics Comparison

Method	PSNR (dB)	SSIM	CNI	Speed (fps)
Hazy Input (Baseline)	12.4	0.61	0.42	—
DCP without Guided Filter	18.7	0.74	0.67	8–10
DCP + Guided Filter (Proposed)	21.3	0.81	0.76	5–7
AOD-Net (Reference, GPU)	23.1	0.85	0.79	15–20

The proposed DCP with guided filter refinement achieves a PSNR improvement of 8.9 dB and an SSIM gain of 0.20 over the unprocessed hazy baseline. The guided filter refinement step contributes an additional 2.6 dB PSNR and 0.07 SSIM improvement compared to the unrefined DCP output, confirming the value of the edge-preserving post-processing step. Comparison with the learning-based AOD-Net reference shows that the proposed physics-based pipeline closes approximately 77% of the quality gap between the hazy baseline and AOD-Net, as measured by PSNR, without requiring any training data or GPU hardware.

5.2 Vehicle Detection Performance

Object detection performance was evaluated by comparing YOLOv8 nano detections on raw hazy frames against detections on dehazed frames, using ground-truth vehicle annotations on the test set. Table IV reports precision, recall, and mean Average Precision at IoU threshold 0.5 (mAP@0.5) for three conditions.

TABLE IV: Vehicle Detection Performance Comparison

Detection Condition	Precision (%)	Recall (%)	mAP@0.5 (%)
Direct on hazy frames	71.2	58.4	63.1
After DCP dehazing (Proposed)	79.8	74.6	76.3
Clear frames (upper bound)	88.1	86.3	87.0

Dehazing preprocessing improves vehicle detection recall by 16.2 percentage points and precision by 8.6 percentage points relative to direct detection on hazy frames. The recall improvement reflects the detector's improved ability to localise vehicles at medium to long range, where atmospheric attenuation in raw foggy frames causes sufficient contrast reduction to suppress detection. Close-range vehicles (within approximately 20 m) are detected consistently in both hazy and dehazed conditions, indicating that short-range attenuation is insufficient to significantly impair the detector. The precision gain suggests that dehazing also reduces false positives, potentially by removing haze-induced bright regions that the detector might incorrectly classify as partial vehicle outlines.

5.3 Qualitative Observations

Visual inspection of paired hazy and dehazed frame sequences at multiple fog densities confirms the quantitative findings. In mildly hazy frames, the dehazing stage restores colour saturation, improves mid-range contrast, and sharpens vehicle silhouettes and road markings. Lane lines that are nearly invisible in raw hazy input are rendered clearly in the dehazed output. In frames with moderate to dense haze, the algorithm removes the whitish veil and reveals scene content not visually perceptible in the original. Some noise amplification is observed in regions of very low transmission, and occasional colour fringing appears at sharp depth discontinuities near the frame periphery where guided filter guidance quality is lower.

5.4 Limitations

The current implementation has several limitations. First, pipeline throughput of 5–7 fps on desktop CPU hardware is below the 25–30 fps standard for surveillance systems, limiting applicability to lower frame-rate monitoring scenarios. Second, the Dark Channel Prior is a statistical assumption that may be violated in scenes dominated by a single bright hue, such as snowfields or deserts; for typical road scenes this concern is less acute due to the diversity of vehicle colours and vegetation. Third, the global atmospheric light estimate A does not account for spatial variation in illumination or haze type across the frame. Fourth, the detection model is applied without domain fine-tuning to dehazed imagery, so residual distributional differences between dehazed outputs and clear training images may leave a performance gap compared to a model fine-tuned specifically on dehazed road imagery.

VI. CONCLUSION

This project successfully developed and validated an integrated pipeline for real-time vehicle detection in foggy road video, combining a physics-based Dark Channel Prior dehazing algorithm with a YOLOv8 nano object detection model. The system is fully implemented in Python using publicly available, well-maintained libraries and operates without the requirement for GPU hardware or labelled training data, making it accessible for deployment in resource-constrained environments. Vehicle detection recall improved by 16.2 percentage points compared to direct detection on hazy frames. These results confirm the fundamental hypothesis that physics-based visibility restoration provides a practical and effective mechanism for improving the performance of standard computer vision models in adverse atmospheric conditions. The guided filter transmission map refinement was shown to contribute substantive quality improvements over the unrefined DCP output, validating the additional computational investment. The selective class filtering of the detection stage ensured focused, relevant output while reducing inference time and display clutter. The side-by-side display of hazy and dehazed-detected frames provides an intuitive visual interface for operators comparing the raw and processed scene representations.

REFERENCES

1. K. He, J. Sun, and X. Tang, "Single Image Haze Removal Using Dark Channel Prior," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 12, pp. 2341–2353, Dec. 2011.
2. K. He, J. Sun, and X. Tang, "Guided Image Filtering," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 6, pp. 1397–1409, Jun. 2013.
3. B. Li, W. Ren, D. Fu, D. Tao, D. Feng, W. Zeng, and Z. Wang, "AOD-Net: All-in-One Dehazing Network," in Proc. IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 4780–4788.
4. S. Sakar et al., "Fog-YOLO: A Deep Learning Model for Object Detection in Foggy Weather Conditions," in Proc. International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 2022, pp. 1182–1188.
5. H. Zhang, V. Sindagi, and V. M. Patel, "Multi-Scale Single Image Dehazing Using Perceptual Pyramid Deep Network," in Proc. IEEE/CVF CVPR Workshops, Salt Lake City, UT, 2018.
6. R. Fattal, "Single Image Dehazing," ACM Transactions on Graphics, vol. 27, no. 3, pp. 1–9, Aug. 2008.
7. S. G. Narasimhan and S. K. Nayar, "Vision and the Atmosphere," International Journal of Computer Vision, vol. 48, no. 3, pp. 233–254, 2002.
8. B. Li et al., "Benchmarking Single-Image Dehazing and Beyond," IEEE Transactions on Image Processing, vol. 28, no. 1, pp. 492–505, Jan. 2019.
9. C. O. Ancuti, C. Ancuti, M. Sbert, and R. Timofte, "Dense-Haze: A Benchmark for Image Dehazing with Dense-Haze and Haze-Free Images," in Proc. IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 2019.
10. Ultralytics, "YOLOv8 Documentation," Ultralytics Inc., 2023. [Online]. Available: https://docs.ultralytics.com
11. G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, 2000. [Online]. Available: https://opencv.org
12. Y. Shao, L. Li, W. Ren, C. Gao, and N. Sang, "Domain Adaptation for Image Dehazing," in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, 2020.
13. Raspberry Pi Foundation, "Raspberry Pi 5 Technical Specification," 2023. [Online]. Available: https://www.raspberrypi.com/products/raspberry-pi-5/
14. 11. Patyrykin, K., & Vasyukova, L. (2025). Environmental Accountability or Symbolic Compliance? A Critical Review of ESG Ratings, Greenwashing, and Indirect Emissions in the Global Insurance Sector. International Journal of Energy Economics and Policy, 15(6), 917–925. https://doi.org/10.32479/ijeep.22770
15. 12. Todupunuri, A. (2024). Explore How AI Can Be Used To Create Dynamic And Adaptive Fraud & Rules That Improve The Detection And Prevention Of Fraudulent & Activities In Digital Banking. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.5014699
16. 13. Poojari, R. (2025). A Comparative Analysis of Fine-Tuning Versus Retrieval-Augmented Approaches for Enhancing Healthcare-Centric Large Language Models.
17. 14. Prodduturi, S. M. K. To Secure Your Paper as Per UGC Guidelines We Are Providing A ElectronicBar code.
18. 15. Reddy, S. K. R. (2024). Designing Blockchain Architecture to Transform Loyalty Rewards into Cryptocurrency Investments.
19. 16. Kalae, U. K. (2023). Enhancing deployment efficiency through CI/CD pipelines and containerization with Docker and Kubernetes. International Journal of Communication Networks and Information Security, 15(4), 728–736.
20. 17. Cyril, H. P., & Kumara, S. Identification of Anomalies via Deep Learning-Based Models for High-Dimensional Telecom Traffic Data.
21. 18. MUDUSU, S. (2025). HEALTH INSURANCE FRAUD DETECTION: THE ROLE OF ADVANCED IT SYSTEMS IN PREVENTING AND IDENTIFYING FRAUD. INTERNATIONAL JOURNAL, 16(1), 3769-3777.
22. 19. Dayal, P. S., Chandra, B. R., Keerthi, M., Sruthi, M., Venkatesh, K., Appalaraju, G., & Eswari, G. (2013). Design of Pyramidal Horn Antenna at 10GHz Using WIPL-D Optimizer. International Journal of Electronics

Communication and Computer Engineering, 4(2).

23. 20. Sruthi, M. V., Sree, V. U., & Soundararajan, K. (2012). Specific removal of motion artifacts in medical image processing. IJECCE, 3(3), 227-229.
24. 21. Santthosh Saai Reddy Purmani. (2026). Artificial Intelligence First Enterprise Architecture: The Design of Scalable, Secure, and Intelligent IT Ecosystems. American Journal of AI Cyber Computing Management, 6(1(2)), 1–8. https://doi.org/10.64751/ajaccm.2026.v6.n1(2).pp1-8
25. 22. GIRISH KOTTE. (2025). ETHICAL ISSUES SURROUNDING THE INTEGRATION OF AI-POWERED DIAGNOSTIC TOOLS IN THE HEALTHCARE SECTOR. American Journal of AI Cyber Computing Management, 5(4), 329–334. https://doi.org/10.64751/ajaccm.2025.v5.n4.pp329-334
26. 23. Viswanathan, V. (2025). Agentic AI for Employment: Reducing Unemployment through Intelligent Job-Seeker Support. LEX LOCALIS–Journal of Local Self-Government.
27. 24. Poojari, R. Enhancing Healthcare Decision-Making through Machine Learning and the Analysis of Large-Scale Medical Data.
28. 25. Kalae, U. K. (2021). Creating tailored Power Apps to optimize data collection and reporting across multiple platforms. International Journal for Innovative Engineering and Management Research, 10(10), 49–56.
29. 26. Viswanathan, V., Shah, A. K., Kubam, C. S., Dontu, S., Gandhi, A., & Singla, P. (2025, August). Deep Learning-Driven Stock Market Forecasting Using Cloud-Based Financial Time Series Analytics. In 2025 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC) (pp. 1-6). IEEE.
30. 27. Patyrykin, K. (2025). CANCEL CULTURE PROBLEM. Lex Localis: Journal of Local Self-Government, 23.
31. 28. Purmani, S. S. R. (2025). Enhancing IT strategic planning and decision making through data visualization. International Journal of Enhanced Research in Management & Computer Applications, 14(4), 75–81
32. 29. Viswanathan, V., Polagani, S. S., Agarwal, R., Akula, S., Dey, S., & Kashyap, R. (2025, September). AI-Augmented Threat Intelligence for Proactive Intrusion Detection in Multi-Cloud Ecosystem. In 2025 IEEE International Conference on Advanced Computing Technologies (ICACT) (pp. 567-572). IEEE.
33. 30. Sruthi, M. V., Soundararajan, K., & Sree, V. U. (2012). Accurate Multimodality Registration of medical images. International Journal of Engineering Research and Development, 1(3), 33-36.
34. 31. Kumara, S. (2026, February). A Lightweight Deep Learning Based Classification Models for Non-Human Identity Threat Detection. In 2026 IEEE 5th International Conference on AI in Cybersecurity (ICAIC) (pp. 1-6). IEEE.
35. 32. Kotte, G. (2025). Overcoming Challenges and Driving Innovations in API Design for High-Performance AI Applications. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.5283649
36. 33. Ranjbareslamloo, S., Dzukeya, G. A., Muhit, M. M. I., & Qattawi, A. (2025). Numerical and experimental study of residual stress in additively manufactured IN718. Manufacturing Letters, 44, 915–927. https://doi.org/10.1016/j.mfglet.2025.915927
37. 34. Viswanathan, V. (2024). Pioneering Ethical AI Integration in Enterprise Workflows: A Framework for Scalable Team Governance. Available at SSRN 5375619.
38. 35. Mudusu, S. K., & Gentyala, S. (2026). Zero-Trust Data Pipelines for AI Systems: A Framework for Secure, Verifiable, and Auditable Data Engineering. JOURNAL OF RECENT TRENDS IN COMPUTER SCIENCE AND ENGINEERING (JRTCSE), 14(2), 10-25.
39. 36. Akhilaiswarya, B., Sree, B. T., Lilly, K., Chowdary, K. H., & Sruthi, M. (2023). Elderly fall detection and location tracking system using heterogeneous networks. Journal of Engineering Sciences, 14(05).