Convergence
A Journal for Young Researchers

ORIGINAL REVIEW

Received: 8-28-25; Accepted: 10-27-25;
Managing Editor: Jonas Katona;
Action Editor: Jonas Katona

Eyes on the Road: Elucidating ViTs and CNNs Under Real-World Noise

Arun Alagappan

M.C.T.M. Chidambaram Chettyar International School, Chennai, India

Abstract

A misclassified citizen or an obstructed traffic light due to image degradation can lead to fatal consequences in autonomous driving systems. Such errors pose a critical threat, as reliability is one of the most debated challenges when it comes to the deployment of these models. These disturbances span from Gaussian blur to extreme lighting contrast. This paper investigates how convolutional neural networks (CNNs) and vision transformers (ViTs) respond to altered visual inputs in autonomous scenarios, drawing on secondary data analysis for evaluations. Additionally, the impacts of noise cleaning techniques on model accuracy and stability are examined. Robustness scores form the core of evaluating reliability in this paper, while interpretability methods such as gradient-weighted class activation mapping (Grad-CAM) and attention maps act as complementary tools that highlight the regions of an image guiding decisions. Benchmark datasets such as Common Objects in Context (COCO) are referenced to ensure fairness in comparison. This methodological approach highlights how traditional metrics like Mean Average Precision (mAP) may conceal critical weak points, and it provides a clearer view of which architectures hold up under perturbations.

Keywords: Convolutional Neural Networks (CNNs), Vision Transformers (ViTs), noise, robustness, grad-CAM, attention

1. Introduction

Autonomous driving models often rely on making split-second choices based on real-time visual data to ensure passenger safety. In an autonomous system, cameras and sensors capture continuous video frames of the surroundings. These frames are processed by algorithms such as Convolutional Neural Networks (CNNs) or Vision Transformers (ViTs) to detect objects, lanes, and signals. The results made by these algorithms guide the control unit to steer, brake, or accelerate. If these predictions fail, they lead to dangerous misclassifications, such as mistaking a pedestrian for a signpost or missing an obstructed traffic light. Failing to make accurate decisions can lead to disastrous misclassifications. One of the main reasons for failure is corrupted visual inputs that contain various forms of perturbations that can influence algorithms (Lambertenghi et al., 2025). Although some autonomous systems also use LiDAR or radar to strengthen their perception,

this paper focuses only on the visual systems where models are still very sensitive to image disturbances. Real-world noise comes from motion blur exposure problems or natural factors like fog and rain. These disturbances make it difficult for the model to stay reliable under unpredictable situations. Despite exponential advancement in prototypes, algorithms still struggle under visual stress, causing poor performance (Bhojanapalli, 2021). This can affect both the safety of the clients and the reliability of the model. The most used architectures in modern systems include Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) (Lai-Dang, 2024). They use distinct mechanisms: CNNs use localized convolutional filters, whereas ViTs induce global awareness through self-attention in their algorithms (Vaswani et al., 2017). These architectures are highly sensitive to the difference between clean versus noisy visual inputs, thereby opening up the doors for misidentification within the models.

Understanding the factors that influence prediction accuracy is fundamental for humans to trust artificial intelligence systems in safety-critical sectors. Interpretability is important because it explains how the model made its decision. In autonomous driving, it helps verify if the model focused on the correct regions before acting, which builds trust and supports safer deployment. Common interpretability techniques use Grad-CAM, which generates heatmaps that emphasize influential regions in an image (Selvaraju et al., 2017). On the other hand, attention rollouts and gradient-based attribution maps are used for ViT-based models due to their complex architecture. These techniques expose significant patches in an image that directly affect the model's response.

Recent studies have proven that ViTs outperform CNNs in perturbation and corrupted settings (Paul & Chen, 2021). However, ViT's interpretability remains restricted, as the attention scores of different visual patches are hard to track through its layer-dependent architecture (Chefer, 2020). On the contrary, CNNs can aid us with more natural clarifications through Grad-CAM. Similar to ViT, under heavy perturbations, even CNNs fail (Bhojanapalli, 2021). This introduces an underexplored gap in the domain that compares interpretability versus accuracy. Both architectures have strengths but are not yet perfect.

In the context of autonomous driving, how do ViTs and CNNs vary in their interpretability and output to perturbed visual inputs, as evaluated primarily through robustness scores with Grad-CAM and attention maps to complete the analysis? In this paper, we explore architectural behavior and understanding by analyzing responses and accuracy with the help of existing empirical papers. The focus of the investigation is to identify each model's accuracy and distortion resilience through the visualization methods mentioned above.

This paper conducts a secondary analysis and a literature review of prior work on CNN and ViT robustness under noise. Observations concluded focus on existing heat maps, model performance, and evaluations under the influence of noise. The analysis in this paper uses the Common Objects in Context (COCO) dataset, which has over 330,000 labelled images across 80 object categories. This allows a fair and consistent comparison of models under the same conditions. Stability of the models under differing levels of perturbations allows simulations of artificial environments (Caesar, 2020; Yu et al., 2020).

Key findings extracted from this investigation will contribute to the development of vision models in the domain not only through their robustness, but also in terms of transparency. This paper provides crucial trade-offs in models built on different foundational structures. The understanding gained from this paper will encourage safety measures to be taken before the deployment of models in this safety-critical field. We also raise attention for the need for robust interpretability

frameworks to mitigate model vulnerabilities. This research supports the deployment of these prototypes by providing an in-depth analysis of the models' behavior to make environments safer with the deployment of the algorithms and the development of trustworthy autonomous systems. The rest of the paper explains the benchmarks and visualization tools used, presents secondary data analysis from existing studies, and ends with findings that show performance differences between CNNs and ViTs under perturbations.

2. Background and literature review

2.1. Computer vision and autonomous vehicles

This section examines how Computer Vision (CV) helps autonomous driving systems process raw sensor data into meaningful information, enabling machines to interpret visuals like humans. It highlights how CV models adapt to environmental changes, strengthen feature extraction, and improve safety in real-world navigation through the integration of software and hardware components (Cordts, 2016; Janai, 2020).

Controlling factors

Core mechanisms that enable the vehicle to drive are controlled by the autonomous system. The decision algorithm is heavily influenced by the objects and entities from the sensor's point of view. Higher-level choices like highway merging, obstacle avoidance, and lane switching are managed by the autonomous modules. (Badue, 2021). The increasing use of artificial intelligence in the transportation field synthesizes a demand for state-of-the-art visual sensors where the input becomes the core influencer of the model's decision. The entire pipeline is interlinked; therefore, any error can affect the reliability of the model. The coordination between these components and the algorithm is crucial in this pipeline, as they are supported by the subsystems. The combination of these modules can predict and plan actions based on the data. Overall, this aids the computational system and provides feedback.

Behavior of self-driving systems

AVs need to perform various complex tasks, including dynamic lane-switching, adapting to traffic, interpreting traffic signals, and avoiding real-time collisions (Levinson et al., 2011). The system uses semantic segmentation and object identification in real-time visuals to identify its surroundings, future trajectories, and path. It also enables pattern identification based on the movement of vehicles to adapt to its environment. These estimations are directly correlated to sensory inputs and the models' ability to interpret them. Industrial applications of these comprehensive models decompose the problem of perturbations into steps and run them through a pipeline of layers. These layers integrate the information gained from the image in predictions of the outcomes. NVIDIA Drive and Waymo have integrated such modular pipelines in their autonomous vehicle systems (Bojarski, 2016).

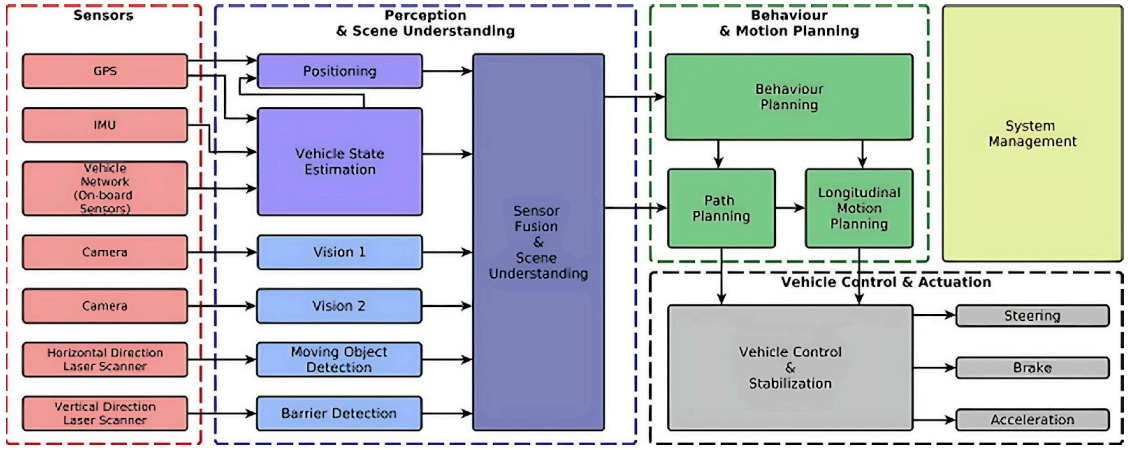

Figure 1: How information moves through an autonomous driving system.

Note: Sensor data from cameras, LiDAR, radar, and GPS are processed through perception modules that use Computer Vision models like CNNs and ViTs. These modules understand the surroundings and send the results to planning and control systems that manage steering, braking, and acceleration in real time (Taş et al., 2016).

Synthesising an intelligent system

Hardware: The requirement for high-end multi-sensors is stressed in the above sections due to their significant influence on the visual models' efficiency. Light Detection and Ranging (LiDAR) sensors that capture depth in pictures, radars for motion detection, and cameras with enhanced ability to capture high-resolution images, benefit the object detection and semantic vision for the models (Levinson et al., 2011). Some limitations faced by these components are restricted processing power, thermal accumulation, and latency. To balance these limitations while achieving maximum efficiency, emerging models as of 2017 suggest architectures like YOLO (You Only Look Once), an object detection model that predicts the location and type of objects in an image in real time. These frameworks neutralize the cons by increasing accuracy under computational limitations (Selvaraju et al., 2017).

Software: To extract meaningful content from the recognition sensors, applications are fused with the workflows to aid the analytic and reasoning process of the model. Methods like HOG, SIFT, and Kalman filters—early methods that identify features and follow object movement in images—were used in general-purpose computer vision tasks to support the algorithm by reducing disturbances in the visual input. These were early solutions to bypass these limitations. Noise, a common term used to describe anomalies in an image, introduces a whole new dynamic sector filled with limitations (Dalal & Triggs, 2005). After recent developments in computer vision, new techniques like deep learning are far superior compared to their ancestors (Khan et al., 2021; K. e. a. He, 2016). Software based on deep learning is embedded into low-level systems to ensure efficiency of the sensors while satisfying memory limits and safety requirements for a trustworthy vehicle (Badue, 2021). The decision regions within CNNs and ViTs are represented through interpretability tools like Attention Maps and Gradient-weighted Class Activation Mapping (Grad-CAM). These mechanisms enhance the transparency and reliability of autonomous systems by making visible the areas that affect the model's prediction under perturbations.

Limitations due to noise: A sensor's perceptions are compromised under natural phenomena like blur, fog, and rain. The input has significant disturbances, potentially covering crucial information for reasonable decisions. For example, when the model is trying to identify traffic lights, a blur caused by swift movements could cause it to overlook the color of the light. These can result in fatal injuries to passengers and reduce the model's accuracy rates (Kalra & Paddock, 2016). Visual distortions, no matter the magnitude, can cause object misclassifications, leading to false reports and plans. Some architectures demonstrate improved robustness to certain distortions; CNNs exhibit resistance to local pixel noise, whereas ViTs perform better against global visual disturbances like occlusions or lighting variations. The increase in performance is due to the extrapolation of the image through global interpretation, allowing these models to understand the "full picture" even under the influence of perturbations (X. Mao et al., 2021; Zhou et al., 2022). The recent evolution has brought the limelight to spread over computer vision, motivating researchers to identify more robust and interpretable models for use within the AV field (H. He et al., 2024; Samek et al., 2015).

Computer vision tasks within autonomous systems

CVs open new doors to take on core driving assignments like lane detection, pedestrian identification, and sign recognition (Janai, 2020). These advantages can boost models to achieve better results when it comes to abiding by the law and the safety of the client, vehicle, and habitat. Similarly, semantic segmentation can acknowledge road elements and provide a descriptive report of the drivable spaces available (Cordts, 2016). Entity awareness and classifications are key to preventing punishable actions and developing the models' insights when it comes to situational awareness (Redmon & Farhadi, 2017).

3. Foundations of AI in Visual Perception Models

This section explains the fundamental principles of artificial intelligence (AI) models that are involved in computer vision tasks. The learning is done through layers, learning plans, and loss evaluation. Layers are like decomposers; they break down an image into different regions and examine them to classify objects and segment elements. CNNs scan the image thoroughly in patches or locally, but ViTs understand the image holistically at a global level (Dosovitskiy, 2020). Loss functions identify differences between the model's output values, providing feedback on the precision of the system such that lower loss means higher performance (Goodfellow et al., 2016). These steps allow CNNs and ViTs to learn from visual information for autonomous vehicles, and with the newer versions of ViT, they are more capable than the older CNNs for processing and interpreting scenes (Touvron et al., 2021; Liu et al., 2021).

3.1. Structural comparison: CNNs and ViTs

This subsection provides an overview of the structural advantages of CNNs and ViTs. Relevant metrics are also elaborated on, such as interpretability, scalability, and resilience to environmental changes. The unique approaches to solving a common problem reveal different perspectives and solutions, causing the dynamic trade-offs between these techniques. Some architectures show improved robustness to specific distortions; CNNs exhibit resistance to local pixel noise, whereas ViTs perform better against global visual disturbances like lighting variations. The increase in performance is due to the holistic processing of the image through global interpretation, allowing these models to understand the "full picture" even under the influence of perturbations. Figure 2 presents the comparison between CNN heat maps and ViT heat maps. Gradient-weighted Class Activation Mapping (Grad-CAM) and Attention maps provide significant insights by generating heatmap overlays on top of images, with varying color intensity to extract the model's focus regions. This method of interpretation can give humans a

deeper understanding of how it reasons and why the system makes particular decisions (Selvaraju et al., 2017). These techniques are emphasized when a model is tested with perturbed visuals. Using the data acquired from the maps can help researchers develop better models that focus on relevant visual elements (H. He et al., 2024).

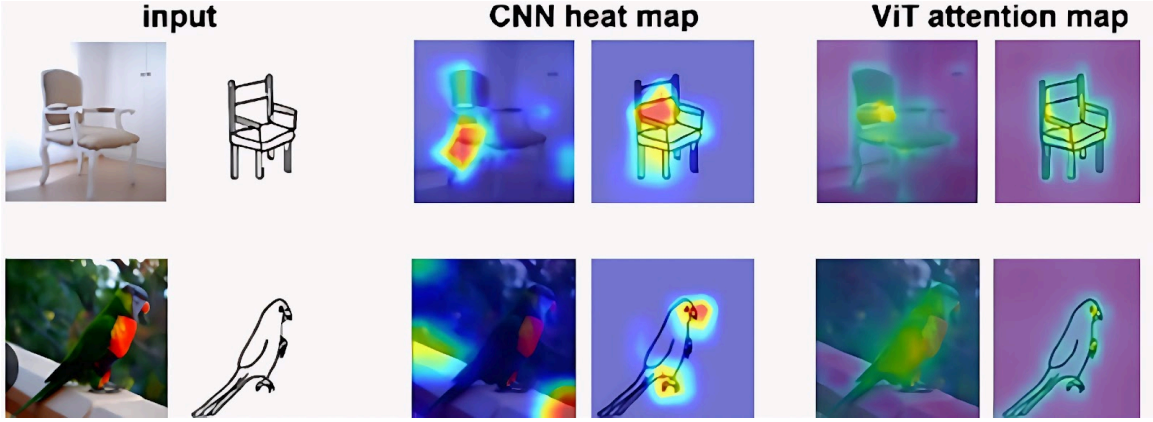

Figure 2: Comparison between CNN and ViT heat maps under perturbed visuals

Note: The upper sequence represents the original inputs, whereas the lower sequence illustrates Grad-CAM (CNN) and Attention Map (ViT) overlays that reveal the guiding regions influencing the model's final decision. (Kang & Seo, 2024)

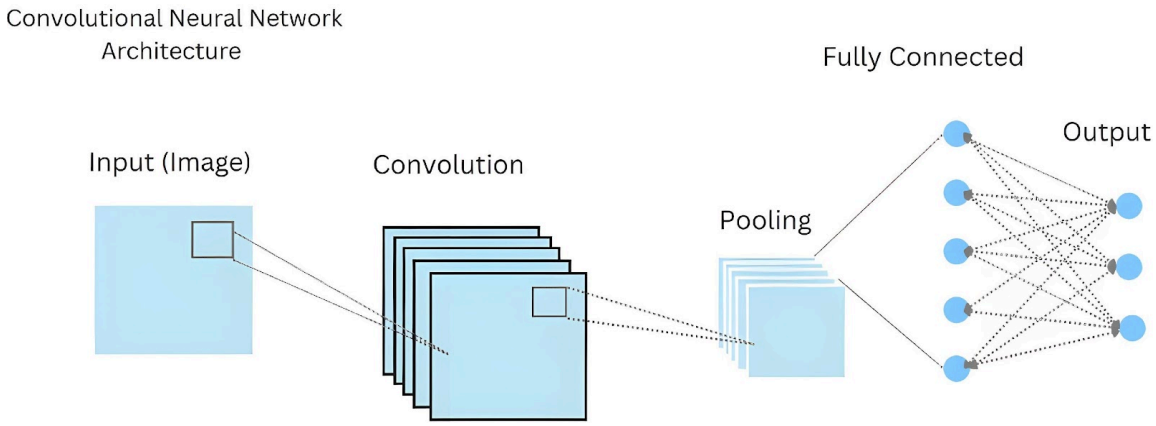

Figure 3: A Convolutional Neural Network (CNN) processes an image through layers of filters, pooling, and connections to recognize patterns and make predictions.

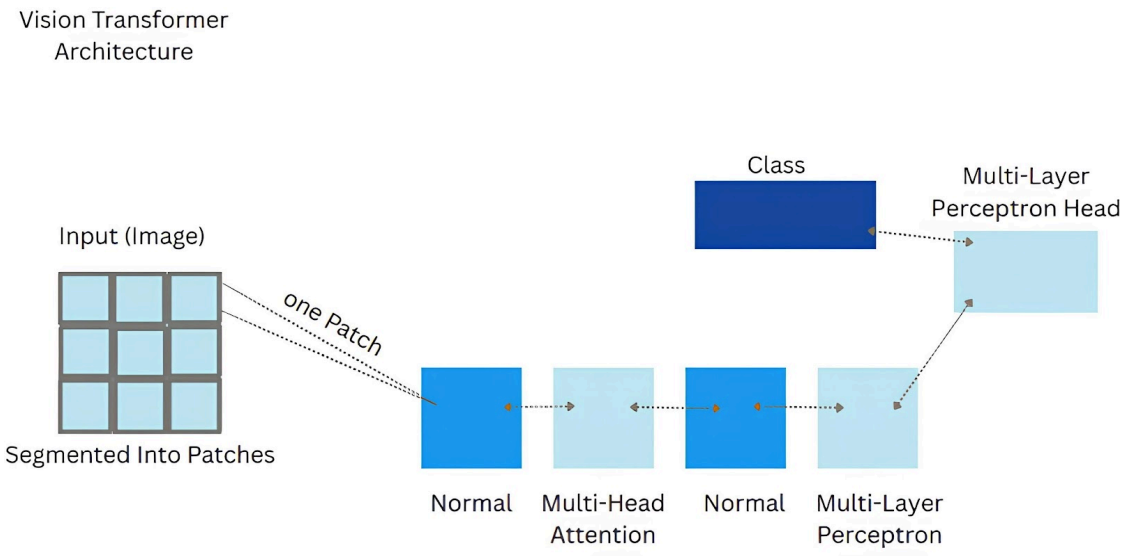

Figure 4: A Vision Transformer (ViT) breaks an image into small patches, uses attention to understand their relationships, and combines the results to classify the image.

Feature processing

Convolutional neural networks (CNNs) use hierarchical layers to extrapolate the model's understanding of the local features present (LeCun, 1998). This method is very effective when it is exposed to corners and textural patterns, as it provides a more extensive interpretation for the model to work with (K. e. a. He, 2016). The strategies used in this architecture align with the needs of embedded systems, making them an ideal candidate for embedded system deployments (Szegedy et al., 2015). Some examples of CNN-based architectures are ResNet, which uses skip connections, a method that allows a model to pass information through layers (K. e. a. He, 2016); RCNN, which uses region-based bounding boxes before classifications for better output (Ren et al., 2015); and YOLO, an object detection system that uses predicting systems in bounding boxes and class probabilities directly from the full image (Redmon & Farhadi, 2017). Figure 3 provides a complete pipeline for CNNs.

On the other hand, ViTs induce a technique that consists primarily of self-attention methods. The fundamental of this technique revolves around the relevance score given to patches of an image. Additionally, each patch is assigned tokens mixed up to synthesize patterns, generally referred to as token mixing. An example of a ViT architecture-based model is DETR (Detection Transformers), a workflow that uses encoders and decoders to help understand images (Carion, 2020). Figure 4 provides a complete workflow for ViTs. These factors provide a robust spatial understanding and global reasoning to the model, enabling it to produce more accurate results (Dosovitskiy, 2020).

Interpretability differences

Inference patterns of different architectures differ under perturbations. This is explored and compared to provide a general overview of the interpretability. CNNs use Grad-CAM, a heatmap that reveals decision-driving regions in an image by tracing activations in the responses (Selvaraju et al., 2017). This interpretation map allows the development of a transparent model

that can be trusted. Correspondingly, ViTs use self-attention maps to show the token dependencies and reasoning abilities of a model. This approach directly impacts the result positively, as it uses the whole image to understand and respond based on context (Chefer, 2021). Overall, ViTs reveal more stable and focused attention under perturbations, demonstrated by the consistent and accurate attention maps under the influence of noise (Chefer, 2021; X. Mao et al., 2021; Zhou et al., 2022). While these interpretability differences are important, robustness metrics remain the primary lens of evaluation in this study.

4. Relevance to interpretability

In various domains, the transparency of advanced computer vision models is gradually decreasing as they become more complex. Interpreting and understanding this void within the models should be considered the most valuable method to develop a truly trustworthy system. In this paper, transparency techniques are considered complementary, providing context to the robustness analysis. Autonomous vehicles are a domain that will be positively affected as transparency in architecture becomes more abundant (Samek et al., 2015). Figure 2 presents the comparison between CNN heat maps and ViT heat maps. Gradient-weighted Class Activation Mapping (Grad-CAM) and Attention maps provide significant insights by generating heatmap overlays on top of images, with varying color intensity to extract the model's focus regions. This method of interpretation can give humans a deeper understanding of how it reasons and why the system made a particular decision (Selvaraju et al., 2017). These techniques are emphasized when a model is tested with perturbed visuals; using the data acquired from the maps can help researchers develop better models that focus on relevant visual elements (H. He et al., 2024).

5. Safety risks in AI-driven AVs

AI-based autonomous vehicle crash data is visualized in Figure 5. The data was collected from public regulatory and investigation reports, such as the NTSB and California DMV records. Only incidents where the autonomous or perception system was directly identified as the main cause were included, while crashes due to human error or unrelated mechanical faults were excluded. This ensures that the numbers strictly represent AI-involved failures.

The data, representing accidents from 2010–2025, shows an increase between 2016–2019 due to large-scale autonomous testing (California DMV, 2022; National Transportation Safety Board (NTSB), 2020). After regulatory reviews and system-level checks were strengthened, it observed a steady decline, leading to a stable and low frequency of accidents. This was due to deeper examination in simulation environments, accurate model testing, and the use of performance metrics like mean average precision before any public deployment. Additionally, the trend in Figure 5 aligns with reported studies that show most AI-related crashes occurred during early testing phases, and were reduced after safety validation was introduced.

Risk factors like adverse weather and low-light contrast demonstrated instability in the model, as some reports mentioned misclassifications under such environmental conditions. Additionally, the fusion of different perspectives provided by sensors (LiDAR, radar, and camera inputs) strengthened object detection reliability, though limitations still existed (Levinson et al., 2011). The first fatality recorded in a fully autonomous system occurred when the Uber ATG algorithm misclassified a pedestrian multiple times within six seconds (National Transportation Safety Board (NTSB), 2019). This led to unstable path predictions and caused the emergency brake system to deactivate due to a false-positive trigger. A pedestrian was killed in Tempe, Arizona, due to the failure of the model's classification techniques. Even though one fatal crash a year may appear statistically low, the event drew heavy public attention and resulted in several new safety regulations and temporary testing suspensions, which collectively reshaped the autonomous vehicle landscape (California DMV, 2022). Fear and caution

developed within the AV space. In the next sections of this paper, we will cover all factors and solutions to prevent fatalities and lower the graph's accident frequency.

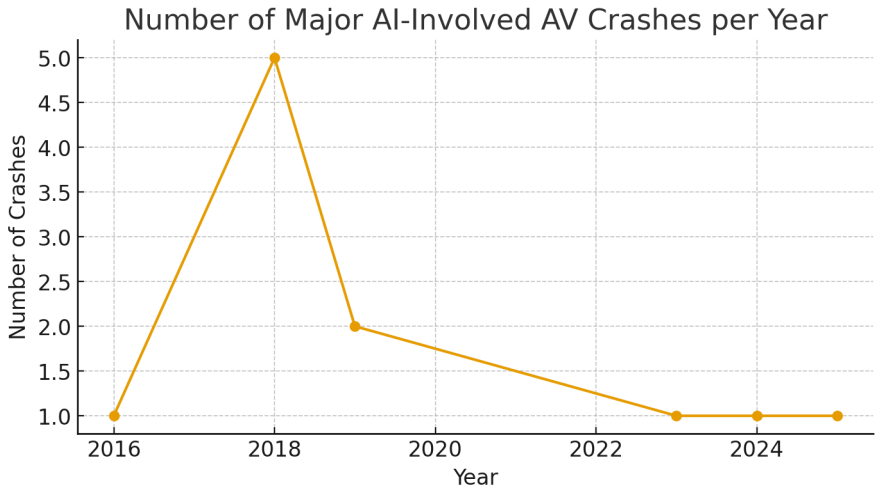

Figure 5: Number of AI-involved AV crashes per year

Note: Representing: Tesla, Williston, Florida (2016); Uber ATG, Tempe, Arizona (2018); Tesla, Mountain View, California (2018); Tesla, Culver City, California (2018); Tesla, South Jordan, Utah (2018); Tesla, Laguna Beach, California (2018); Tesla, Delray Beach, Florida (2019); Tesla, Gardena, California (2019); Cruise Robotaxi, San Francisco, California (2023); Waymo, San Francisco, California (2024); Zoox, Las Vegas, Nevada (2025).

6. Types of noise

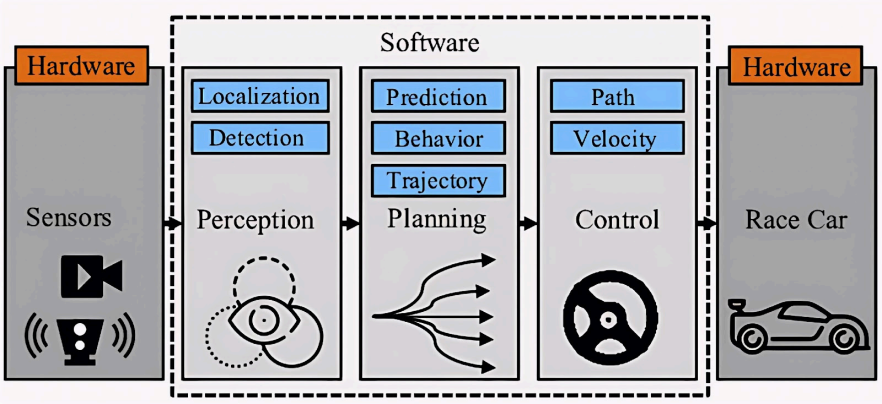

Figure 6: A simplified perception pipeline representing how an input image from the LiDAR, radar, or camera travels through a model (CNN or ViT), with layers that allow for noise mitigation. Additionally, this figure describes the process of decoding an image in three sections: perception, planning, and control. (ResearchGate, 2025)

During training, noise is intentionally injected into the dataset as data augmentation to ensure that the model can handle real-world perturbation variations (Shorten & Khoshgoftaar, 2019). To be specific, CNNs use preprocessing filters such as de-noising to improve feature stability (Zhong et al., 2017; Krizhevsky et al., 2012). On the other hand, ViTs use patch-level normalization and adaptive positional encoding to work around this roadblock (Dosovitskiy, 2020; Touvron et al., 2021). When it comes to deployment, sensor fusion helps reduce the impact of noise in perception modules (Levinson et al., 2011, C. Zhang et al., 2020). Preprocessing will remove environmental distortions, ensuring that visual inputs are consistent. This is crucial, as both ViTs and CNNs depend on clean images.

6.1. Motion and Gaussian blur

Blurs have the ability to hide or distort crucial information, and such a lack of clarity in vision can allow models to overlook key aspects that influence the interpretation. Swift movements can cause motion blur (see Figure 7). These blurs decrease the interpretable information available for the model (Nah et al., 2017). Similarly, Gaussian blurs are synthetic simulations of elements that can distract a model, and are often used in benchmark datasets to analyze situation-based noises (Hendrycks & Dietterich, 2019a).

Figure 7: Motion and Gaussian blur distort a picture. Fast movement or synthetic blur hides important details and confuses the model when detecting objects. (AIEase, n.d.).

6.2. Fog and haze

Fog adds scattered lighting, creating luminous areas that can wash out color and contrast in an image. Figure 8 provides a visual for how much detail this perturbation can remove (Sakaridis et al., 2018). Detecting entities can become difficult under this type of noise, and dehazing is necessary to make data predictable (Li et al., 2017). AOD-Net techniques provide a great network to prevent this type of error, allowing for quick and easy haze removal (Li et al., 2017).

Figure 8: Fog removes color and contrast, washing out the picture. The model loses small details and can miss objects on the road.

6.3. Rain & atmospheric noise

As mentioned in the previous section, a very common and natural noise is rain. Rain creates long, high-frequency streaks and varied lighting on artifacts. Figure 9 depicts noise caused by rain (Lr-ps, 2017). Detailed restoration networks can mitigate these disturbances, and cleansing techniques can preserve content while removing rain or any weather-based anomalies (Fu, 2017). Rain-specific training allows models to ignore streaks in the image during classification (H. Zhang et al., 2019).

Figure 9: Rain creates streaks and light changes that block vision. The blurred parts make it harder for the model to see objects correctly.

6.4. Occlusion & clutter

Objects blocking the vision of our human eyes restrict the obtainable information. Similarly, when cars or any entity block a sector of an image, anything behind the figure is hidden. Figure 10 provides a description and results when occlusions are present (Ryu & Chung, 2021). This can prevent models from getting a comprehensive understanding of the context presented, and contributes to object misclassification (Hendrycks & Dietterich, 2019a). Training models with synthetic occlusion can develop resilience within the model's algorithm. It prevents overfitting, making it a necessity for every model, as it has the potential to exponentially enhance the model's interpretability (Ghiasi, 2018).

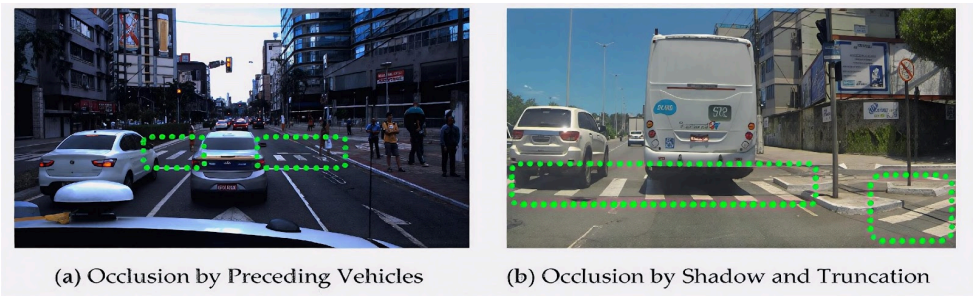

Figure 10: An example of how objects can block what is behind them. Occlusion confuses the model by cutting off parts of the image.

6.5. Illumination changes

Sudden sun glares can blind vision detectors, or night scenes can confuse the model due to the drastic illumination percentages in different patches of the image. Figure 11 shows a low-contrast environment, where the loss of clarity and detail is portrayed by the change in lighting (Edwards, 2025). This noise mainly affects texture-sensitive models (Hendrycks & Dietterich, 2019b). To combat this, the addition of preprocessing methods like exposure corrections and dynamic range compensation is necessary (C. Zhang et al., 2020).

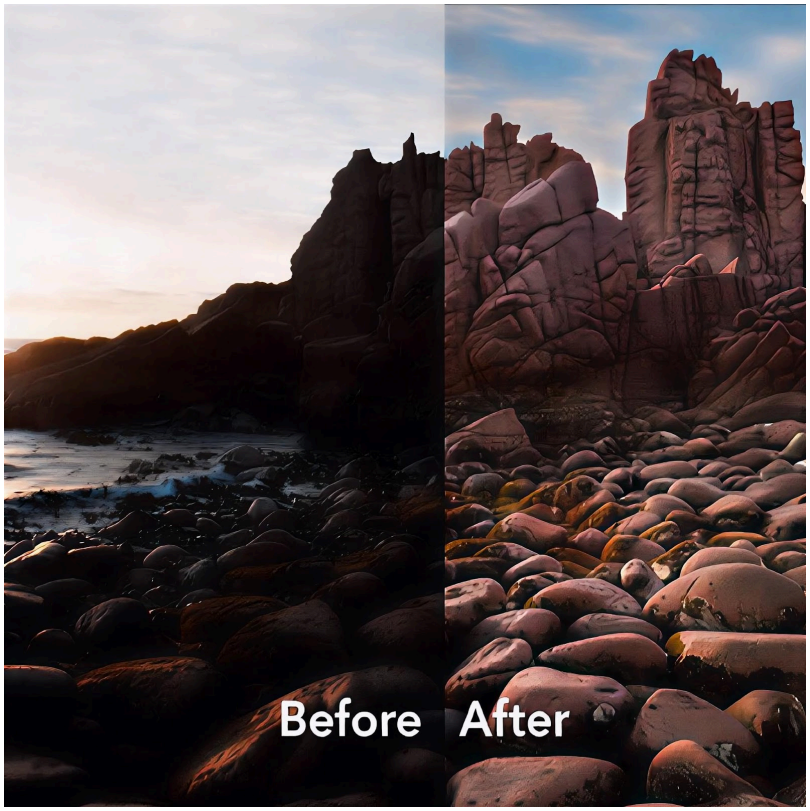

Figure 11: Lighting changes between bright and dark areas confuse the model. Loss of texture and contrast reduces what the model can detect.

7. Common techniques for noise mitigation

In real-world autonomous driving scenarios, pristine images are rare to come by. Noise in the form of blurry lighting, weather, and sensor limitations heavily influences the model's interpretability. This section addresses solutions to these imperfections, such as methods like pixel-level cleaning, deblurring, visibility enhancements, weather-specific processing, and robust training to mitigate these perturbations. These approaches collectively improve clarity and preserve detailed information for answering prompts. Additionally, they develop resilience against perturbations in images.

7.1 Pixel-level cleaning

Images tend to have different variations of light contrast, which can cause dynamic changes in interpretation, as it makes it hard for models to split images into layers and understand the whole picture. The usage of histogram equalizations or Contrast Limited Adaptive Histogram Equalization (CLAHE) mitigates these risks. They spread out pixel intensity values to ensure darker regions become brighter and lighter regions become dimmer. This improves the overall contrast of the picture, giving the model a more genuine perspective of the image (Zuiderveld, 1994). The size of the image also links to the response;

providing uniform inputs (e.g., 640 x 480, 1024 x 576) can increase the model's understanding. This increase is present while the model compares sizes of objects to synthesize patterns, successfully adding to the model's predictions and output (Howard, 2017).

7.2. Deblurring and visibility enhancement methods

Blurs and visibilities have affected human sight, causing inaccurate judgments and a lack of understanding of surroundings. Figure 12 provides an example by comparing a clean and a perturbed image. The loss of details is significant in the image, resulting in poor accuracy in models. A model's vision is a downgraded version of the human eye when it comes to instantaneous recognition; therefore, these are considered perturbations, and they must be cleaned before sending it through the interpretation phase. Multi-Scale Convolution-Deblurring Networks is an efficient yet simple method to decrease the blurred regions present in the visual provided. It breaks down and downsamples the resolution of the image, making it easier to correct large blurs. Once it attains the lowest possible resolution, the layer progressively corrects Gaussian or motion blurs. Additionally, the model gains invaluable insights for classification, making it easier to predict a blurred image versus a natural image (Nah et al., 2017). All-in-One Dehazing (AOD) is a technique used to cleanse a picture from perturbations in a singular and contained space. Figure 13 offers a graphic illustration of the benefit of utilizing an AOD net. This method's efficiency is emphasized under foggy or polluted conditions of the environment, as it aids in the restoration of a sharp image (Li et al., 2017).

7.3. Weather-specific and rain removal pre-processing

Raindrops create streaks in images, acting as disturbances and causing unwanted lines within a visual. This negatively impacts the model's stability. Therefore, techniques such as the Detail-Recovery Network (DNN) are employed to reduce rain's significance and eradicate its effect. DNNs use high-frequency isolations, where rain streaks usually appear, to remove these stripes. High-pass filters are applied to input images to carry out this procedure, and continuous training on these specific perturbation models develops resilience. Loss functions are integrated to balance rain removal and detail preservation, which can help preserve fine edges in a figure (Fu, 2017). Other weather-based anomalies in an image are treated using weather-augmented training. This allows models to handle any impact caused by the environment. The training involves synthetic additions to the image to reduce the fragility of the model. Moreover, it improves the robustness of the domain for the models, allowing systems to adapt to dynamic conditions (H. Zhang et al., 2019).

Figure 12: Motion and Gaussian blur ("What Is Imerge Pro?" Manula, FXhome, 2021)

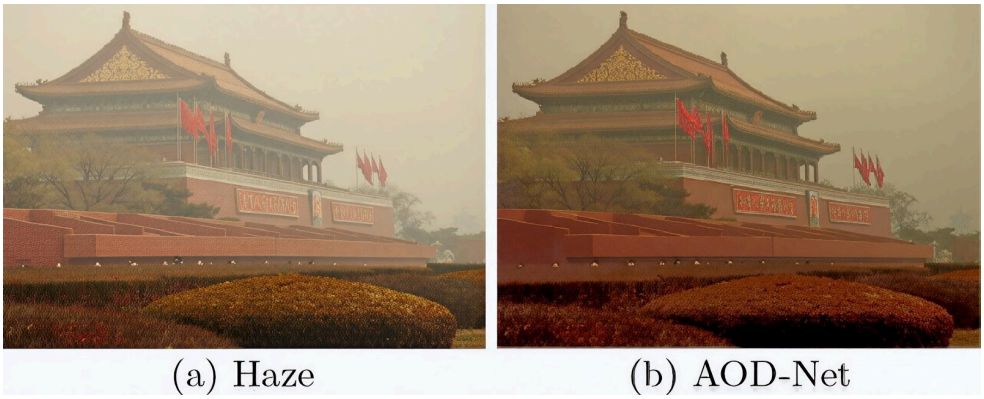

Figure 13: AoD motion and Gaussian blur removal

8. Robust training strategies

Algorithms often find shortcuts to reduce work by memorizing patterns and replicating them when prompted. This is a frequent issue across all sectors. To avoid this overfitting, randomly assigned databases are used in the learning and testing stages of the development cycle. Domain-particular approaches revolve around noise-specific enhancement. The approach involves intentionally including individual perturbations like Gaussian blur, motion blur, and fog. By exposing the algorithm to these conditions, it will learn to adapt and analyze rather than retain or memorize the patterns. A large-scale yet robust dataset or benchmark for synthetic analysis is ImageNet-C (Hendrycks & Dietterich, 2019a). Although adding anomalies strengthens the model's interpretability, it fails to provide statistical evidence to fully address this concept's validity. A more

developed approach involves weaving corrupted and clean images in the training batch, preventing overfitting and degradation of the model's performance in all stages. Additionally, empirical evidence from studies on semantic segmentation under adverse weather conditions shows that this method stabilizes feature recognition across the domains (Michaelis et al., 2019).

9. Model strengths across perturbation scenarios

The diversity of the vision model architecture allows for specific segmented deployment based on each model. In this section, the main factor analyzed is noise in relation to the model's performance, as certain algorithms provide better values under some types of noise. In Table 1, the CNN architecture is able to provide stability under moderate rain and fog. On the other hand, ViTs were able to stand their ground when introduced to low-light glare, shadows, and pixel noise. The hybrid systems could preserve efficiency when introduced to granular-level noise like dust and snow, and were able to adapt to partial occlusions. Strengths include the following: stability ensures reliable and balanced performance, retention provides durable and strong memory, and robustness reflects the adaptability of the algorithm. Coverage delivers the wide scope of the analyzed system, detection enables consistent recognition, resilience shows the flexible and persistent nature of the models, confidence refers to trustworthiness, and adaptability ensures versatile and evolving capacity.

Table 1: Model strengths across different perturbation scenarios

Perturbation Type	Scenario	Best Model	Strength	Citations
Environmental	Moderate rain	CNN	Stability	Fu, 2017; H. Zhang et al., 2019
Environmental	Fog	CNN	Retention	Li et al., 2017; Sakaridis et al., 2018
Environmental	Low-light glare	ViT	Robustness	Chen et al., 2018; Zhou et al., 2022
Natural	Snow	Hybrid CNN+ViT	Coverage	Michaelis et al., 2019; Wang et al., 2024
Natural	Dust	Hybrid CNN+ViT	Detection	Hendrycks & Dietterich, 2019a; X. Mao et al., 2021
Natural	Shadows	ViT	Resilience	Chefer, 2021
Adversarial	Pixel noise	ViT	Confidence	X. Mao et al., 2021; Paul & Chen, 2021
Occlusion	Partial blockage	Hybrid CNN+ViT	Adaptability	Ryu & Chung, 2021

10. Observation & Analysis

This section aims to provide a secondary analysis of CNN and ViT-based model precision under the influence of natural or synthetic noise. This paper does not deploy or train models directly; instead, it performs a secondary analysis of existing empirical results drawn from prior studies on CNNs and ViTs. The focus is on interpreting the observed robustness trends and visual explanations rather than replicating experimental trials. Additionally, it provides perspectives on the effect and mitigation strategies used to clean perturbations. This is a crucial factor for building robust and interpretable models. The COCO (Common Objects in Context) dataset is a benchmark dataset containing over 200,000 labeled images across 80 different everyday object categories. It consists of both indoor and outdoor scenes with various lighting and weather conditions, making it suitable for analyzing vision models' response to real-world perturbations (Lin et al., 2014; C. Mao et al., 2023). It is the most used dataset for evaluating object segmentation and detection. Additionally, it forms the basis for all Mean Average Precision (mAP) evaluations used in this study. The perturbations examined include Gaussian blur, motion blur, fog, haze, rain, illumination changes, partial occlusion, and pixel noise (Hendrycks & Dietterich, 2019a; C. Mao et al., 2023), covering both environmental and synthetic distortions typically encountered in road scenes.

10.1. Perturbation Effects on Model Performance

As discussed previously, the impact on models when introduced to perturbation is significant, and this is shown in the output of Table 2. The models selected for comparison—Faster R-CNN, Mask R-CNN, EfficientDet-D4, DETR, Deformable DETR, DINO, and Swin-L—were chosen based on their leading benchmark architectures for object detection (Carion, 2020; Ren et al., 2015; Zhou et al., 2022). Each model captures a different design: region-based proposals (CNNs) against transformer-based global attention (ViTs). This enables a balanced cross-architecture analysis.

Table 2: Perturbation effects on model performance on the COCO dataset

Model	Type	Perturbation	Clean mAP (%)	Perturbed-mAP (%)	Drop(%)	Source
Faster R-CNN	CNN	Natural	62.8	48.8	14.0	Shankar et al., 2021
Mask R-CNN	CNN	Natural	63.1	49.4	13.7	Shankar et al., 2021
EfficientDet-D4	CNN	Natural	49.4	~35	14.4	Zhou et al., 2022
DETR	ViT	Natural	42.0	~32	10.0	C. Mao et al., 2023
Deformable DETR	ViT	Natural	45.4	~34	11.4	C. Mao et al., 2023
DINO (Swin-L)	ViT	Natural	51.3	~38	13.3	Zhou et al., 2022
Swin-L Detector	ViT	Natural	58.7	~44	14.7	Zhou et al., 2022

10.2. Mitigation and Robustness Strategies in CNNs & ViT

Cleaning the noise before inputting the image into the algorithm is the most logical way to preserve the accuracy of the model. Mean Average Precision (mAP) was used as the primary evaluation metric, as it combines recall and precision into a single score, accounting for both detection accuracy and completeness (Hendrycks & Dietterich, 2019a; C. Mao et al., 2023). It also enables a fair comparison between architectures and is most commonly used in object detection benchmarks like COCO. Cleaning can be done at different levels: architecture, dataset, and interpretation maps. The architectural approach involves upgrading CNNs with deformable convolution for adaptive receptive fields (Dai, 2017). The preparation phase, where models can be trained on mixed atmospheric datasets, is the specific focus of the dataset method. The resilience of the model is further enhanced by incorporating artificial noise, such as blur, fog, and noise distortion, into visual inputs (Shorten & Khoshgoftaar, 2019). Using interpretable tools such as Grad-CAM and attention maps, which assist in identifying attention trigger marks under perturbation, is another complementary approach (Samek et al., 2015). This allows diagnosis of specific vulnerability areas in perception-based models.

None of the evaluated models in this comparison incorporate the mitigation strategies detailed earlier (deblurring, histogram equalization, or weather-specific preprocessing) (Shankar et al., 2021; Zhou et al., 2022). Hence, performance drops reflect raw model vulnerability without external robustness reinforcement. These algorithms, such as Faster R-CNN and DETR, are primarily meant for research and are not directly deployed in commercial autonomous systems. However, their detection modules serves as the foundation of real-world perception models used by industry systems like Waymo and NVIDIA Drive (Bojarski, 2016; Levinson et al., 2011).

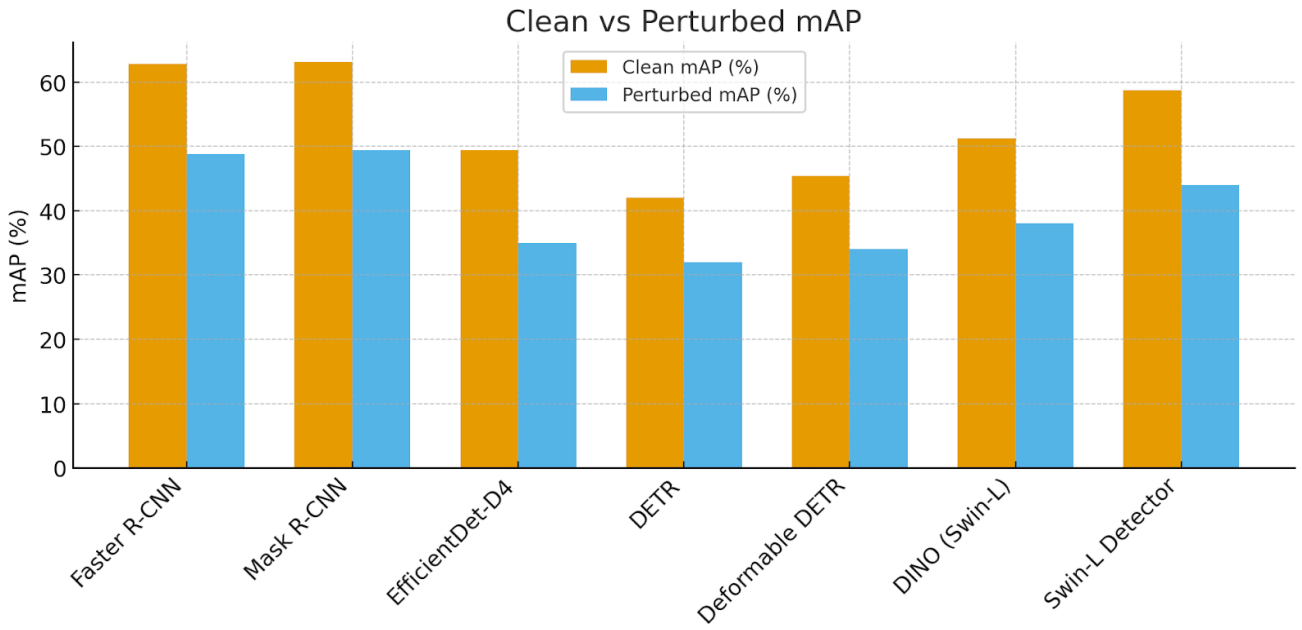

Figure 14: Model accuracy on clean images versus noisy images. Every model drops in performance when exposed to natural distortions

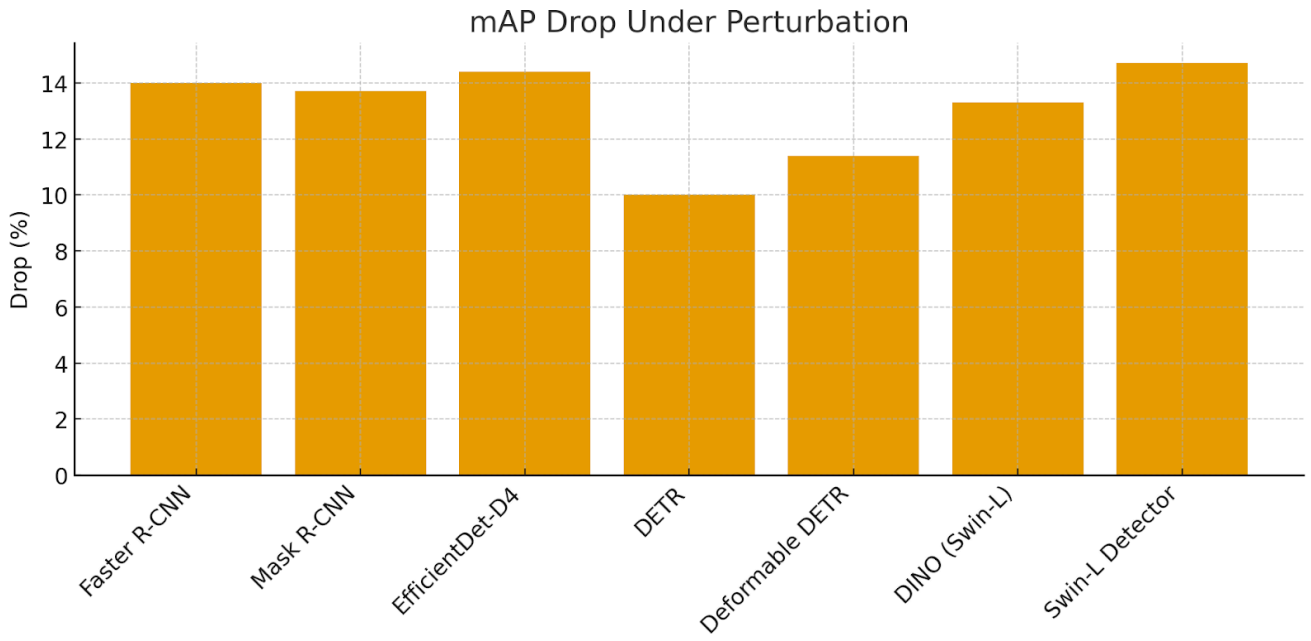

Figure 15: How much each model's accuracy falls because of noise. ViTs lose less accuracy than CNNs, proving stronger stability.

10.3. Key Observations & Analysis

Natural perturbations like blur, lighting variation, and weather distortions significantly degrade the detection accuracy of both CNN and Vision Transformer models on the COCO dataset. There can be improvements in cleaning techniques and resilience to noise, as all of the models listed in Table 2 show a significant drop in performance.

Performance drop across all models

Every model, no matter the architecture, faces a decline in Mean Average Precision (mAP) when introduced to naturally disturbed images, varying from 10.0% (DETR) to 14.7% (Swin-L Detector). This analysis confirms that robustness to real-world problems remains a common shared vulnerability for both architectures.

CNN models: robustness in certain scenarios

Faster R-CNN and Mask R-CNN retain over 48% mAP when exposed to perturbations. Although their drop rates are relatively high—14% and 13.7%, respectively—these models are able to maintain about 50% accuracy. These scores are the highest among the 8 models selected for the analysis (see Table 2). EfficientDet-D4, despite having a lower clean mAP (49.4%), is able to retain 35% accuracy while demonstrating toughness when compared to more powerful CNN models. The above graph indicates that robustness is not directly proportional to performance, making it a challenge to draw conclusions from the graph.

Mixed robustness patterns

DETR demonstrates the smallest performance drop (10.0%), indicating a stable feature extraction even under anomalies in the image. As a face vault, this model performs well, but other ViT models like Swin-L are able to yield much better results under perturbed and clean images. This indicates a trade-off: DETR offers more consistent performance across clean and perturbed conditions, whereas Swin-L prioritizes maximum performance, despite a slightly larger performance loss when facing noise. The varying drop sequences highlighted by the graph suggest that certain models like DETR focus on preserving stability, whereas systems like Swin-L rely on peak gains.

Higher accuracy does not guarantee robustness

Although Swin-L detectors have a clean mAP of 58.7%, it falls to ~44% under defects, which proves that a higher baseline precision does not refer to better perturbation resistivity. Similarly, another model with a similar issue is EfficientDet-D4, where it holds a modest clean mAP yet suffers a drop compared to the state-of-the-art CNNs and ViTs. This emphasizes that architectural and noise-handling approaches, rather than raw precision, are the deciding factors for robustness in real-world implications.

Role of perturbation type

All results in Table 2 corresponded to natural perturbations like motion blur and fog, ensuring domain-based outcomes within the secondary analysis. These perturbations tend to affect texture-dependent models more in fine-detail recognition tasks while impairing ViTs' ability to model global information. All noise-disturbances analyzed are based on realistic driving conditions such as fog, rain, glare, and occlusion (Hendrycks Dietterich, 2019; Lambertenghi et al., 2025). They are either captured in COCO's natural imagery or simulated using methods that are validated in prior autonomous studies, ensuring their realism.

11. Advanced Robustness Analysis

Traditional graphs (Figures 14, 15) illustrate clean vs perturbed mAP, providing an understanding of the "average" performance. While these insights are helpful, these face-value perspectives hide critical weak points in the algorithm. A model can appear strong overall, yet crumble under corruptions in an image. This is an important constraint for AV tasks because the model deployment is delayed by rare but harmful failures (Michaelis et al., 2019), (Wang et al., 2024).

11.1. Multi-Metric Robustness Framework

Table 3: Reliability distributions for CNN-based models (Faster R-CNN, Mask R-CNN, EfficientDet-D4) and ViT-based models (DETR, Deformable DETR, DINO, Swin-L). Values are derived from benchmark robustness studies.

Algorithms	μ (mean-rPC)	σ (stability)	min-rPC (worst-case)	Source
Faster R-CNN	0.77	0.06	0.58 (snow)	Wang et al., 2024

Mask R-CNN	0.78	0.05	0.61 (fog)	Michaelis et al., 2019
EfficientDet-D4	0.71	0.08	0.55 (motion blur)	Wang et al., 2024
DETR	0.76	0.04	0.60 (jpeg)	C. Mao et al., 2023
Deformable DETR	0.75	0.05	0.57 (defocus)	C. Mao et al., 2023
DINO (Swin-L)	0.74	0.07	0.59 (frost)	Zhou et al., 2022
Swin-L Detector	0.75	0.06	0.62 (contrast)	Zhou et al., 2022

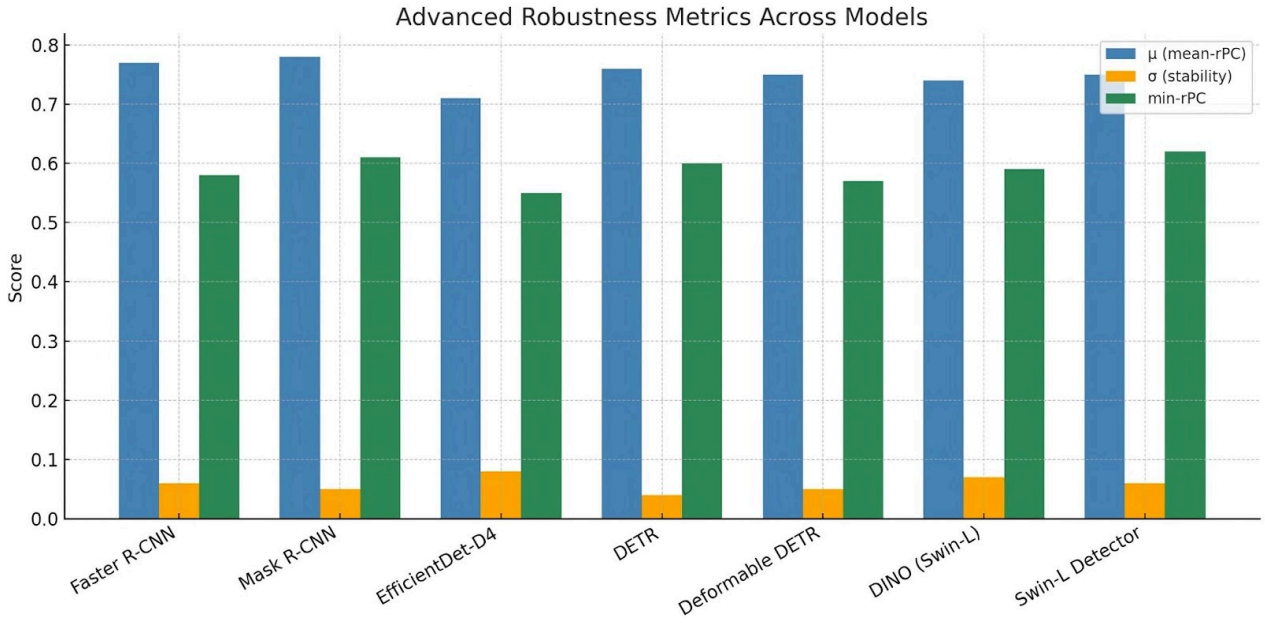

Figure 16: Three robustness metrics for all models. CNNs score higher on average strength, while ViTs show more stability and better worst-case results.

To tackle the issue, durability is evaluated on three useful metrics: mean rPC (μ) offers the average robustness across natural corruptions, stability (σ) displays variations across corruptions, demonstrating consistency, and min-rPC reveals the worst-case robustness, highlighting hidden weaknesses in the algorithms.

Mask R-CNN shows a higher mean robustness (μ = 0.78) and accuracy (σ = 0.05) when compared to Faster R-CNN and EfficientDet-D4. By comparison, it outperforms Faster R-CNN in worst-case robustness by 0.03, proving its supremacy in the design. Its balance of accuracy and dependability makes it the most reliable CNN contender.

The Swin-L Detector shows a significantly higher mean robustness ($\mu = 0.75$) with good σ (0.06), proving its overall dominance in the ViT architecture. Also, it has the best worst-case consistency (min-rPC = 0.62) across the 8 models listed above. The algorithm outperforms DETR and DINO by combining uniformity with excellent lower-limit security, making it the best candidate for the most reliable model in this specific architecture.

While CNNs remain relatively stronger in mean robustness, the Swin-L Detector proves that ViTs can exceed CNNs in worst-case safety assurances, which is more essential for autonomous driving tasks.

12. Discussion and Conclusion

The comparison between Vision Transformers (ViTs) and Convolutional Neural Networks (CNNs) lays out a direct understanding of how these architectures vary in terms of interpretability, robustness, and their universal reliability in autonomous systems. CNNs present a higher ability when it comes to recognizing local textures and features in detailed and structured environments (K. e. a. He, 2016; LeCun, 1998). This ability helps them achieve strong performance values when exposed to stable inputs. However, they fail to maintain this performance when the disturbance affects the whole image. Conditions like fog, rain, and low-light contrast create visible degradation that directly impacts their stability (X. Mao et al., 2021).

On the other hand, ViTs are built to understand images globally through self-attention mechanisms (Zhou et al., 2022; Dosovitskiy, 2020). This gives them better results under noise and real-world visual perturbations, but their computational cost and lower interpretability make them difficult to apply in real-time and resource-driven systems. These results connect with the earlier studies that show the trade-off between interpretability and robustness (Chefer, 2021; Samek et al., 2015). CNNs build more interpretable Grad-CAM heatmaps that identify the model's region of interest within an image, while ViTs depend on attention patches that are disbanded and difficult to trace. This difference lowers their transparency in decision-making.

The reliance on datasets like COCO (Lin et al., 2014) also becomes a concern because they do not fully reflect unpredictable conditions faced in the real world (Michaelis et al., 2019; Wang et al., 2024). Benchmark datasets often provide clean or synthetic data, which exaggerates stability and does not represent the unpredictable nature of real-world noise. Hardware efficiency also plays a key role and is often overlooked. ViTs may show more strength under distortions, but they need large amounts of computational power and resources, making them inefficient in edge or embedded systems. CNNs, on the other hand, remain more suitable for cost-limited and time-sensitive systems due to their low power use and compact design. These findings align with Paul & Chen (2021), who similarly observed that Vision Transformers demonstrate greater resilience to distributional noise than traditional CNN-based architectures.

Robustness is not only about how strong the model is, but also how efficient the entire pipeline functions. The sensors, preprocessing quality, and synchronization between systems all shape how stable the output remains. The next step in this domain must move away from comparing architectures in isolation and shift toward hybrid perception frameworks that bring the best qualities of both. Combining the local recognition ability of CNNs with the global reasoning of ViTs can help models stay strong under different distortions. The system should not depend only on the camera; sensors like LiDAR and radar must work together with visual algorithms (Prakash et al., 2021). This kind of integration can improve the model's awareness and reduce the risk of failure during visual stress.

A dataset that captures real-world conditions is necessary because synthetic distortions do not represent how models actually perform in natural environments. Real samples collected from real scenarios can show the true reaction of the model under unpredictable noise. They can also reveal the weak points that benchmark data usually hide. There must be one consistent and clear method to evaluate results so that all models are compared equally and the outcomes remain reliable. An important observation from this study is that normal evaluation methods, such as Mean Average Precision (mAP), are not enough to show how the model reacts under real stress. They only present the average accuracy, hiding internal inconsistencies.

Advanced metrics such as mean-rPC, stability, and minimum robustness coefficient (min-rPC) give a better and more complete idea of the model's true stability. These parameters measure more than just accuracy; they show how stable and reliable a model remains when it faces visual noise. They reflect how well the model can handle strong distortions and reveal the weakest performance level it can drop to. This advanced way of evaluating performance presents a clearer picture of a model's behavior under real-world pressure, which is more valuable for safety-critical applications.

Overall, these findings show the architectural and technical gaps that must be addressed before achieving a completely interpretable and robust system. CNNs and ViTs both contribute strongly in different areas, but individually, they cannot guarantee consistency or trustworthiness. CNNs deliver transparency and low energy use but are vulnerable to distortions. ViTs perform more stably under various noises but lack clarity and require heavier computation. The results of this study, such as the 10% mAP drop for DETR and 13–14% accuracy loss in Faster R-CNN and Mask R-CNN, confirm that high accuracy alone does not define robustness. Hybrid frameworks that combine both architectures and include cross-sensor coordination with advanced evaluation methods can create more reliable and understandable autonomous systems. Bringing together interpretability and resilience builds a strong base for safe and flexible AI-driven vehicles that can maintain stability even under unpredictable environmental changes.

13. Methods

13.1. Overview

This section explains the approach and reasoning used to analyze the performance of vision models under visual disturbances. Rather than building or deploying models, this study focuses on secondary analysis, which involves extracting existing benchmark data and comparing the outcomes. This approach avoids experimental bias and reveals a broader and more reliable view, since the values are taken from peer-reviewed empirical papers rather than single controlled runs.

13.2. Evaluation framework

The evaluation follows a standardized statistical framework designed to measure the stability and dependability of each model when it is introduced to distortions. For this purpose, three robustness parameters were calculated: mean robustness (μ), stability (σ), and minimum robustness coefficient (min-rPC). They were derived using the following formulations:

Mean robustness (μ)

As defined in Equation 1, μ (mean-rPC) represents the average robustness. It ranges from 0 to 1, and the higher the score, the more performance it maintains under perturbation (Yi et al., 2021).

$$\mu = \frac{1}{N}\sum_{i=1}^{N}\frac{mAP_i - mAP_{clean}}{mAP_{clean}} \quad (1)$$

Stability (σ)

According to Equation 2, σ (stability) refers to the variation in robustness. The range of this metric is usually from 0 to 0.2. The lower the value, the more stable the model will be (Dong, 2025).

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{mAP_{perturbed,i}}{mAP_{clean}} - \mu\right)^2} \quad (2)$$

Minimum robustness coefficient (min-rPC)

Equation 3 defines min-rPC (worst-case), portraying the lowest score obtained by a model under any sort of noise. It ranges from 0 to 1. The higher the value, the better, resulting in fewer failures (Subbaswamy & Saria, 2021).

$$\min - rPC = \min\left(\frac{mAP_{perturbed,i}}{mAP_{clean}} - \mu\right) \quad (3)$$

Here, μ measures how much accuracy a model can retain across all noise types, σ captures the steadiness of that performance, and min-rPC identifies the lowest reliability point under extreme corruption. Together, they represent both average stability and the model's weakest link, something single-metric evaluations like plain mAP cannot show.

13.3. Computation procedure

Each model's clean and perturbed accuracy values were obtained from verified sources. The computation procedure involved four sequential steps:

1. Collecting clean and corrupted mAP results from benchmark datasets
2. Normalizing results across the architectures for consistent comparison
3. Computing μ, σ, and min-rPC for each natural perturbation type
4. Collating all values into a unified reliability distribution table for CNNs and ViTs

This framework is superior to traditional accuracy-based comparison because it captures both consistency and failure tolerance. Rather than judging models by one final score, it evaluates how they behave under stress, revealing their hidden weaknesses and stability margins. This layered method allows fair, architecture-agnostic analysis, making the outcomes both transparent and scientifically reproducible.

All calculations were carried out in Python 3.10 using the NumPy and Pandas libraries for statistical data processing. Additionally Matplotlib was used to create the robustness plots. Equations 1, 2, and 3 have been directly implemented into the Python scripts to maintain consistency between datasets, and all of the computations involved have been cross-checked for reproducibility.

13.4. Analysis of accident data

These robustness trends are put into context with real-world outcomes by collecting accident report data from publicly available records of autonomous vehicles over the period from the initial introduction of AI-driven perception systems to their current deployment. The data was organized in chronological order and visualized on a timeline graph, which expressed the frequency and distribution of the reported accidents across different time frames. This shows the correlations of advances in vision model architectures with incident trends, visually summarizing how improvements in the reliability of perception models fall in line with the historical progress of integrating AI into vehicles.

13.5. Model processing pipeline

All the models used in this study follow one process that starts with the input image, goes through their layers, and ends with evaluation outputs. The preprocessing phase involves resizing and normalizing the visual input to make brightness, contrast, and scale uniform. CNN-based architectures such as Faster R-CNN, Mask R-CNN, and EfficientDet-D4 use this step to stabilize inputs before the convolutional pipeline (K. e. a. He, 2016; Shankar et al., 2021; Zhou et al., 2022), while Vision Transformers such as DETR, Deformable DETR, and Swin-L divide the image into fixed patches that are converted into tokens with positional encodings to retain spatial order (Carion, 2020; Liu et al., 2021; C. Mao et al., 2023).

CNNs pull out features via sequential convolution and pooling, while RPN layers separate objects and BiFPN combines features across scales (Michaelis et al., 2019; Wang et al., 2024). Transformer models, on the other hand, utilize self-attention to obtain global relations, with DETR employing an encoder–decoder architecture, Deformable DETR optimizing attention for irregular areas, and Swin-L combining local and global information via shifted-window attention (Liu et al., 2021; Zhou et al., 2022). The models were tested on COCO with comparable noise conditions (Hendrycks & Dietterich, 2019a; Lin et al., 2014), in addition to robustness being calculated using mAP, μ, σ, and min-rPC. Research following 2017 indicates that the incorporation of CLAHE, AOD-Net, or Multi-Scale Deblurring networks is able to further enhance clarity, robustness, and interpretability (Li et al., 2017; Nah et al., 2017; Selvaraju et al., 2017).

13.6. Literature Screening Process

The studies used in this secondary analysis were filtered through a structured literature screening process. Searches were conducted using Google Scholar, IEEE Xplore, arXiv, and SpringerLink, with keywords such as “CNN robustness,” “Vision Transformer perturbations,” “autonomous driving vision,” and “noise mitigation.” Only peer-reviewed preprints between 2017 and 2025 reporting mAP scores or robustness metrics were included. This review prioritized papers that used standardized benchmarks, such as COCO or ImageNet-C, and clearly specified noise types and evaluation metrics. This ensured methodological consistency across the compared results and minimized dataset or reporting bias.

14. References

Badue, C., Guidolini, R., Carneiro, R. V., Azevedo, P., Cardoso, V. B., Forechi, A., Jesus, L., Berriel, R., Paixão, T. M., Mutz, F., de Paula Veronese, L., Oliveira-Santos, T., & De Souza, A. F. (2021). Self-driving cars: A survey. *Expert Systems with Applications*, 165, Article 113816. https://doi.org/10.1016/j.eswa.2020.113816

Bhojanapalli, S., Chakrabarti, A., Glasner, D., Li, D., Unterthiner, T., & Veit, A. (2021). Understanding robustness of transformers for image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 10231–10241). https://doi.org/10.1109/ICCV48922.2021.01007

Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U., Zhang, J., Zhang, X., Zhao, J., & Zieba, K. (2016). *End-to-end learning for self-driving cars* (arXiv:1604.07316). arXiv. https://doi.org/10.48550/arXiv.1604.07316

Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., & Beijbom, O. (2020). nuScenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 11621–11631). https://doi.org/10.1109/CVPR42600.2020.01164

California Department of Motor Vehicles. (2022). *Autonomous vehicle disengagement reports 2022*. https://www.dmv.ca.gov/portal/vehicle-industry-services/autonomous-vehicles/disengagement-reports/

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020). End-to-end object detection with transformers. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds.), *Computer Vision – ECCV 2020* (pp. 213–229). Springer. https://doi.org/10.1007/978-3-030-58452-8_13

Chefer, H., Gur, S., & Wolf, L. (2021). Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 782–791). https://doi.org/10.1109/CVPR46437.2021.00084

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., & Schiele, B. (2016). The Cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3213–3223). https://doi.org/10.1109/CVPR.2016.350

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., & Wei, Y. (2017). Deformable convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 764–773). https://doi.org/10.1109/ICCV.2017.89

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (Vol. 1, pp. 886–893). https://doi.org/10.1109/CVPR.2005.177

Dong, Z., Wang, Y., Sun, X., Liu, D., Han, G., & Liu, Y. (2025). Tire slip angle estimation-based lateral stability control strategy for trajectory tracking scenarios of distributed drive autonomous electric vehicles. *Control Engineering Practice*, 156, Article 106297. https://doi.org/10.1016/j.conengprac.2024.106297

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*. https://openreview.net/forum?id=YicbFdNTTy

Fu, X., Huang, J., Zeng, D., Huang, Y., Ding, X., & Paisley, J. (2017). Removing rain from single images via a deep detail network.

In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3855–3863). https://doi.org/10.1109/CVPR.2017.186

Ghiasi, G., Lin, T.-Y., & Le, Q. V. (2018). DropBlock: A regularization method for convolutional networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 31). Curran Associates. https://proceedings.neurips.cc/paper/2018/hash/7edcfb2d8f6a659ef4cd1e6c9b6d7079-Abstract.html

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. https://www.deeplearningbook.org

He, H., Wu, W., Wang, L., Zhang, Y., & Liu, J. (2024). CF-CAM: Cluster filter class activation mapping for reliable gradient-based interpretability. *Pattern Recognition*, 150, Article 110387. https://doi.org/10.1016/j.patcog.2024.110387

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778). https://doi.org/10.1109/CVPR.2016.90

Hendrycks, D., & Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*. https://openreview.net/forum?id=HJz6tiCqYm

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). *MobileNets: Efficient convolutional neural networks for mobile vision applications* (arXiv:1704.04861). arXiv. https://doi.org/10.48550/arXiv.1704.04861

Janai, J., Güney, F., Behl, A., & Geiger, A. (2020). Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends in Computer Graphics and Vision*, 12(1–3), 1–308. https://doi.org/10.1561/0600000079

Kalra, N., & Paddock, S. M. (2016). *Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability*? RAND Corporation. https://doi.org/10.7249/RR1478

Kang, S., & Seo, K. (2024). Sketch classification and sketch based image retrieval using ViT with self-distillation for few samples. *Journal of Electrical Engineering & Technology*, 19, 4441–4450. https://doi.org/10.1007/s42835-024-01889-6

Khan, S., Naseer, M., Hayat, M., Zamir, S. W., Khan, F. S., & Shah, M. (2022). Transformers in vision: A survey. *ACM Computing Surveys*, 54(10s), Article 200. https://doi.org/10.1145/3505244

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems* (Vol. 25, pp. 1097–1105). Curran Associates. https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html

Lai-Dang, T. (2024). Interpretable medical imagery diagnosis with self-attentive transformers: A review of explainable AI for health care. *BioMedInformatics*, 4(1), 113–126. https://doi.org/10.3390/biomedinformatics4010008

Lambertenghi, G., Antonello, R., Corno, M., & Savaresi, S. M. (2025). Benchmarking image perturbations for testing automated driving assistance systems. In *Proceedings of the IEEE International Conference on Software Testing, Verification and Validation* (pp. 246–257). https://doi.org/10.1109/ICST62969.2025.00034

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324. https://doi.org/10.1109/5.726791

Levinson, J., Askeland, J., Becker, J., Dolson, J., Held, D., Kammel, S., Kolter, J. Z., Langer, D., Pink, O., Pratt, V., Sokolsky, M., Stanek, G., Stavens, D., Teichman, A., Werling, M., & Thrun, S. (2011). Towards fully autonomous driving: Systems and algorithms. In *Proceedings of the IEEE Intelligent Vehicles Symposium* (pp. 163–168). https://doi.org/10.1109/IVS.2011.5940562

Li, B., Peng, X., Wang, Z., Xu, J., & Feng, D. (2017). AOD-Net: All-in-one dehazing network. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 4770–4778). https://doi.org/10.1109/ICCV.2017.511

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), *Computer Vision – ECCV 2014* (pp. 740–755). Springer. https://doi.org/10.1007/978-3-319-10602-1_48

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., & Guo, B. (2021). Swin Transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 10012–10022). https://doi.org/10.1109/ICCV48922.2021.00986

Mao, C., Jiang, L., Dehghani, M., Vondrick, C., Sukthankar, R., & Essa, I. (2023). COCO-O: A benchmark for object detectors under natural distribution shifts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 20316–20327). https://doi.org/10.1109/ICCV51070.2023.01864

Mao, X., Qi, G., Chen, Y., Li, X., Duan, R., Ye, S., He, Y., & Xue, H. (2021). Towards robust vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 12042–12051). https://doi.org/10.1109/CVPR46437.2021.01186

Michaelis, C., Mitzkus, B., Geirhos, R., Rusak, E., Bringmann, O., Ecker, A. S., Bethge, M., & Brendel, W. (2019). Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv*. https://doi.org/10.48550/arXiv.1907.07484

Nah, S., Kim, T. H., & Lee, K. M. (2017). Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3883–3891). https://doi.org/10.1109/CVPR.2017.35

National Transportation Safety Board. (2019). *Collision between vehicle controlled by developmental automated driving system and pedestrian, Tempe, Arizona, March 18, 2018* (Highway Accident Report NTSB/HAR-19/03). https://www.ntsb.gov/investigations/AccidentReports/Reports/HAR1903.pdf

Paul, S., & Chen, P.-Y. (2022). Vision transformers are robust learners. In *Proceedings of the AAAI Conference on Artificial*

Intelligence (Vol. 36, No. 2, pp. 2071–2081). https://doi.org/10.1609/aaai.v36i2.20103

Prakash, A., Chitta, K., & Geiger, A. (2021). *Multi-modal fusion transformer for end-to-end autonomous driving* (arXiv:2104.09224). arXiv. https://doi.org/10.48550/arXiv.2104.09224

Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, faster, stronger. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 7263–7271). https://doi.org/10.1109/CVPR.2017.690

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 28, pp. 91–99). Curran Associates. https://proceedings.neurips.cc/paper/2015/hash/14bfa6bb14875e45bba028a21ed38046-Abstract.html

ResearchGate. (2025). *Autonomous vehicles on the edge: A survey on autonomous vehicle racing – scientific figure* [Figure]. https://www.researchgate.net/figure/Autonomous-vehicles-on-the-edge_fig1_361208596

Ryu, S.-E., & Chung, K.-Y. (2021). Detection model of occluded object based on YOLO using hard-example mining and augmentation policy optimization. *Applied Sciences*, 11(15), Article 7093. https://doi.org/10.3390/app11157093

Sakaridis, C., Dai, D., & Van Gool, L. (2018). Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, 126(9), 973–992. https://doi.org/10.1007/s11263-018-1072-8

Samek, W., Binder, A., Montavon, G., Lapuschkin, S., & Müller, K.-R. (2016). Evaluating the visualization of what a deep neural network has learned. *IEEE Transactions on Neural Networks and Learning Systems*, 28(11), 2660–2673. https://doi.org/10.1109/TNNLS.2016.2599820

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 618–626). https://doi.org/10.1109/ICCV.2017.74

Shankar, V., Roelofs, R., Mania, H., Fang, A., Recht, B., & Schmidt, L. (2021). Evaluating machine accuracy on ImageNet. In *Proceedings of the 38th International Conference on Machine Learning* (pp. 9588–9598). PMLR. http://proceedings.mlr.press/v139/shankar21a.html

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6, Article 60. https://doi.org/10.1186/s40537-019-0197-0

Subbaswamy, A., & Saria, S. (2020). From development to deployment: Dataset shift, causality, and shift-stable models in health AI. *Biostatistics*, 21(2), 345–352. https://doi.org/10.1093/biostatistics/kxz041

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1–9).

https://doi.org/10.1109/CVPR.2015.7298594

Taş, Ö. Ş., Kuhnt, F., Zöllner, J. M., & Stiller, C. (2016). Functional system architectures towards fully automated driving. In *Proceedings of the IEEE Intelligent Vehicles Symposium* (pp. 304–309). https://doi.org/10.1109/IVS.2016.7535379

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., & Jégou, H. (2021). Training data-efficient image transformers & distillation through attention. In M. Meila & T. Zhang (Eds.), *Proceedings of the 38th International Conference on Machine Learning* (pp. 10347–10357). PMLR. http://proceedings.mlr.press/v139/touvron21a.html

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 30, pp. 5998–6008). Curran Associates. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

Wang, S., Li, Y., Liu, Y., Tian, Q., Li, C., Wang, D., & Li, X. (2024). *OmniDrive: A holistic LLM-agent framework for autonomous driving with 3D perception, reasoning and planning* (arXiv:2405.01533). arXiv. https://doi.org/10.48550/arXiv.2405.01533

Yi, C., Yang, S., Li, H., Zhou, X., & Wu, B. (2021). *Benchmarking the robustness of spatial-temporal models against corruptions* (arXiv:2110.06513). arXiv. https://doi.org/10.48550/arXiv.2110.06513

Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., & Darrell, T. (2020). BDD100K: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 2636–2645). https://doi.org/10.1109/CVPR42600.2020.00271

Zhang, C., Yan, Q., Zhu, Y., Li, X., Sun, J., & Zhang, Y. (2020). *Attention-based network for low-light image enhancement* (arXiv:2005.09829). arXiv. https://doi.org/10.48550/arXiv.2005.09829

Zhang, H., Sindagi, V., & Patel, V. M. (2019). Image de-raining using a conditional generative adversarial network. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(11), 3943–3956. https://doi.org/10.1109/TCSVT.2019.2920407

Zhong, Z., Zheng, L., Kang, G., Li, S., & Yang, Y. (2020). Random erasing data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 34, No. 7, pp. 13001–13008). https://doi.org/10.1609/aaai.v34i07.7000

Zhou, W., Zhang, Z., Zhang, D., Luo, C., & Zhang, T. (2022). *Understanding the robustness in vision transformers* (arXiv:2204.12451). arXiv. https://doi.org/10.48550/arXiv.2204.12451

Zuiderveld, K. (1994). Contrast limited adaptive histogram equalization. In P. S. Heckbert (Ed.), *Graphics gems IV* (pp. 474–485). Academic Press. https://doi.org/10.1016/B978-0-12-336156-1.50061-6

Acknowledgements

I would like to thank Dr. Krishnan for teaching me how to structure and develop a research paper. His lessons on academic writing and organization helped me understand how to build my work independently. I also want to thank my TA, Samuel, for his very useful comments that guided me in refining my writing and improving the overall clarity of my paper.

I would also like to thank my colleagues for their encouragement regarding the initial steps of this project and for sharing their thoughts and ideas with me. Our discussions have made me see new perspectives and further improved the way I present my ideas.

I also want to thank my parents for always being supportive and providing all the facilities that I needed to accomplish this research. Their patience and motivation continue to drive me. Finally, I would like to thank my brother, who helped and participated during my work, especially when I needed feedback and technical details.

Author Biography

During his middle school years, **Arun Alagappan** developed an interest in logical thinking, computer science, and mathematics. He also took an MIT machine learning course to broaden his understanding of artificial intelligence systems. He took a Python and database (Postgres SQL) course to learn the basics of artificial intelligence in order to delve deeper into this field. He became very interested in pursuing computational and artificial intelligence research as a result. Due to the field's rapid development, he discovered in the tenth grade that he was passionate about conducting research projects. Outside of the classroom, he shows dedication and teamwork by playing basketball for his school and winning two interschool tournaments. By working with his peers to create a working model for computer vision systems, he actively participates in the computer science department.

Mentor Contribution Statement

Dr. Karthik Krishnan, professor at Northeastern University, was my mentor and played an important role in helping me understand how to build and organize a research paper. He did not take part in writing or editing this paper, but he taught me the essential skills needed to create it on my own. He guided me through the research paper formats, how each section flows, and how to make the ideas transition clearly from one part to another. He described how to articulate a research question and how to follow the argument with clarity and supportive evidence. He also aided me in the process of reading and breaking down other research papers to understand how professional studies are written and how they connect their methods and findings. These lessons gave me the tools to design my own study, organize my thoughts, and write in a proper academic format. In the process of writing, Dr. Krishnan provided me with feedback concerning structure and clarity, which guided me to think critically and improve my writing skills. His role was that of teaching, not contributing to this paper's content. Everything in this manuscript, from the ideas, analysis, writing, and presentation, was completed by me with the feedback provided by Dr. Krishnan and his teaching assistant, Samuel. The knowledge and foundation I received from them helped me to work independently and complete this paper with a strong understanding of academic research and writing.