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

[redacted]

Abstract

A misclassified citizen or an obstructed traffic light due to image degradation can lead to fatal consequences in autonomous driving systems. Such hindrances 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. Hence, this paper explores how convolutional neural networks and vision transformers respond to altered visual inputs in autonomous scenarios. To understand a model's focus, we propose analyzing the attention or focus of a model under the influence of perturbed images. Additionally, we investigate the impacts of noise cleaning techniques on the model and its stability by using metrics like mAP. Robustness scores forms the core for evaluating reliability in this paper, while interpretability methods such as Grad-CAM and attention maps are complementary tools that reveal guiding regions in an image that influence decisions. Utilizing COCO, a standardized benchmark dataset, enables a fair study of these models. This approach will reveal the best models in each architecture after careful comparison using the above methods.

Keywords: CNN, ViT, Self-attention, noise, perturbation, nuScenes, visualisation, Grad-CAM

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

Autonomous driving models often rely on making split-second choices based on real-time visual data to ensure passenger safety. If they fail to make accurate decisions, it will 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, 2025). Real-world noise ranges from motion blurs to sensor noises which can be caused by swift movement during the exposure time. Natural noises like fog, rain, and blurs are common types of perturbation that autonomous driving models face daily. 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, 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 way of predicting the accuracy is fundamental for humans to trust artificial intelligence systems in safety-critical sectors. Common interpretability techniques use Grad-CAM, which generates heatmaps that emphasize the influential regions in an image [(Selvaraju, 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 affects 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 their own 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? Therefore, 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 as mentioned above.

This paper conducts a secondary analysis and a literature review of prior work on CNN and ViT robustness under noise. Observations concluded are focused on existing heat maps, model performance, and evaluations under the influence of noise. The analysis process involves visualizing key findings through graphs for fair evaluations. Visual data used in prior studies will go through comparisons to obtain usable figures and trends in models. The referenced data that these models will be tested on are COCO benchmark datasets; occasionally real-world noise is used too. Stability of the models under differing levels of perturbations allows simulations of artificial environments [(Caesar, 2020; Yu, 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 deployment of models in this safety-critical field. It also raises attention to 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 enable such environments to become safer with the deployment of the algorithms and the development of trustworthy autonomous systems.

Computer Vision and Autonomous Vehicles

This segment studies the use of Computer Vision (CV) in perspective-driven autonomous driving systems; the model typically converts raw sensor inputs into structured and valuable data. Computer vision allows machines to understand and interpret visual data; it enables computation systems to emulate the human visual system through advanced technologies. CV models enable robust object detection and environmental understanding, resulting in better decision-making support under perturbations. Additionally, CV models are trained to adapt to surroundings like changes in the weather and lighting, providing more powerful feature extraction abilities. Thereby improving safety in navigation tasks in real-world descriptions [(Cordts, 2016; Janai, 2020)]. This section explores the various components that a synthetic system can control in the modernized transport systems. Furthermore, it delves into the link between the software and hardware components within this autonomous pipeline.

Controlling Factors

Core mechanisms that enable the vehicle to drive are controlled by the autonomous system. These components include steering, braking, signaling, indicators, honking, lighting, and throttle [(Thrun, 2010)]. These mechanisms are commanded using the decisions made by the system. The decision algorithm is heavily influenced by the objects/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; 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.

Behaviour 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, 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 movement of vehicles to adapt to its environment; these estimations are directly correlated to the sensory inputs and the models' ability to interpret. Industrial applications of these comprehensive models decompose the problem into steps and run them through a pipeline of layers. These layers integrate this information in forecasts and forethoughts of the outcomes. NVIDIA Drive and Waymo have integrated modular pipelines in their autonomous vehicle systems [(Bojarski, 2016)].

How Modules Collaborate

This section outlines the sensing and perception pipelines that underscore autonomous driving systems, exploring how sensor data is processed through recognition layers to make real-time decisions. The suggested pipeline consists of LiDAR, radar, and camera sensors for the visual inputs, as this is connected to the object detection architecture. These recognition architectures primarily consist of Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs). These frameworks will be discussed in detail throughout the paper. Once the model understands the image, it will extrapolate features to form the path and the behavior of the AV. This is directly fed to the hardware to change variables such as steering and acceleration [(Prakash, 2021)]. These intelligent systems induce suggestions to upgrade algorithms and pattern recognition. The reduction of concentrated workloads on one sector of the model enables a more accurate and precise decision. The individual layers and components that share its workload are able to extract patterns and adapt accordingly to enhance split-second decision-making.

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 models' efficiency. LiDAR sensors (Light Detection and Ranging), which are used for capturing depth in the pictures. Radars for motion detection and cameras for their enhanced ability to capture high-resolution images. These sensors benefit the object detection and semantic vision for the models [(Levinson, 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 and ViT. These frameworks neutralize the cons by increasing accuracy under computational limitations [(Selvaraju, 2017)].

Software

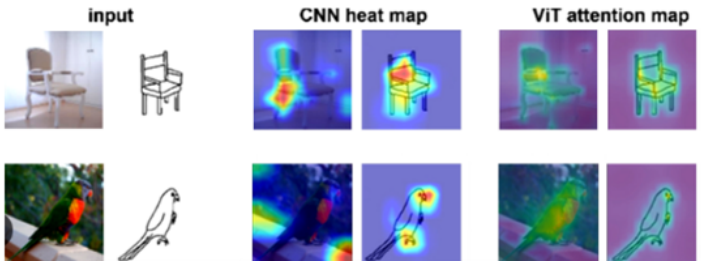
To extract meaningful content from the recognition sensors, applications are fused with the workflows to aid the analytic and reasoning process of the model. Methods like HOG, SIFT, and Kalman filters were used in general-purpose computer vision tasks to support the algorithm by reducing disturbances 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 development in computer vision, new techniques like deep learning are much superior compared to their ancestors. These mechanisms primarily consist of CNNs for semantic segmentation and object detection and ViTs for interpreting complex situations [(Khan, 2021; Vaswani, 2017)]. Software 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)].

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 the traffic lights, a blur caused by

Figure 1

Input and heat map comparison [(Kang & Seo, 2024)]



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 system architectures have demonstrated significant improvements and stability under these noise types. Further research into this drawback has led to the conclusion that CNNs tend to be more stable against adversarial attacks,whereas ViTs are often more stable against natural corruptions and occlusions due to their ability to leverage global image context. 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. e. a. Mao, 2021; Zhou, 2022)]. The recent evolution has brought the limelight to spread over computer vision; this motivates researchers to identify more robust and interpretable models for use within the AV field [(H. e. a. He, 2024; Samek, 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 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, 2017)].

Classical versus Deep Learning Approaches

Traditional computer vision heavily relied on manual feature classification like SIFT or ORB; these techniques are effective under structured and standardized environments. Unfortunately, when they are introduced to noise, they seem to crumble and become inaccurate [(Dalal & Triggs, 2005)]. More advanced and current models engage in these tasks with CNNs; the ability to learn hierarchical features from datasets provides a significant advantage. It has the capability to solve robust classification and detection problems [(K. e. a. He, 2016)]. Another deep learning architecture is ViT. This newly developed algorithm is able to reason with global features while maintaining performance under perturbations [(Naseer, 2021)].

Relevance to interpretability

The lack of transparency increases every day in advanced CV models no matter the domain. Interpreting and understanding this void within the models is the most efficient method to develop a truly trustworthy system. In this paper transparency techniques are considered complementary, providing context to the robustness analysis. Autonomous vehicles is a domain that will be positively affected as transparency in an architecture becomes more abundant [(Samek, 2015)]. Advancements in CV model's lucidity ensure traceability and trust within clients, additionally offering a more reliable and consistent model. Figure 1. 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 colour 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, 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. e. a. He, 2024)]. This method of gaining clarity

within these models contributes to the development and the usability of these advanced computational techniques.

Foundations of AI in Visual Perception Models

This section explores the basic principles behind Artificial Intelligence (AI) models that are involved in computer vision. The learning and development happen through layers, learning methods, and loss evaluation. These are mechanics that support CNNs and ViTs in interpreting visual data for autonomous vehicles.

Layers

Layers are like decomposers; they break down an image into different sectors and analyze them to identify objects and segment parts. Their tasks can vary from basic edge separation to element detection. CNNs do a detailed scan of the image in parts or locally. On the contrary, ViTs perceive the image by understanding the whole image at a global stage. These dynamic differences in the architecture allow a wide range of outputs that these models exhibit.

Gradient Descent

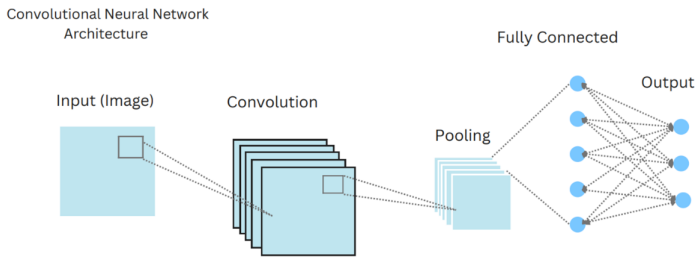
This is effectively a feedback system where models learn from their errors by comparing outputs against expected results. After each prediction, the model makes small adjustments to its internal parameters to reduce future mistakes and misclassifications. This continuous adjustment process allows the model to steadily refine its pattern recognition and improve its inference over time.

Loss Functions

This algorithm measures the variation between the model's predicted values and the expected values. This acts as a measurement tool for trained models; they provide extremely helpful

Figure 2

Convolution Neural Network architecture



insights into the precision rates of the system. Lower loss shows that the model is performing better or as expected. This is a major step that all models go through before deployment to verify that balance and precision are up to state-of-the-art standards.

Show of Knowledge

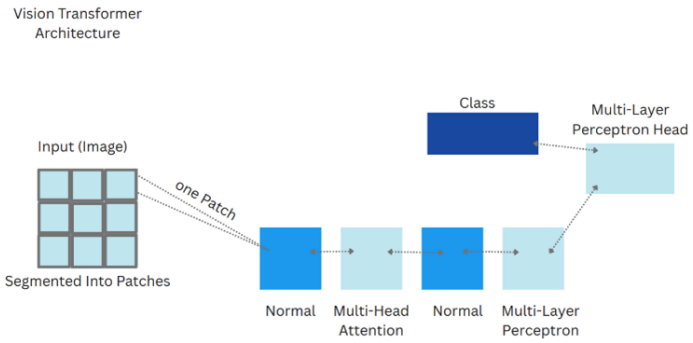
CNNs and ViT frameworks use these basic graphs and values to improve the models' performance during training. Due to ViTs' recent growth, they are able to handle and provide significantly more accurate answers for complex scenes than the older CNN models. These evolution metrics are standardized as benchmarks for these algorithms for comparative and comprehensible purposes.

Structural Comparison: CNNs and ViTs

This section 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 solve a common problem expose different perspectives and solutions, causing the dynamic trade-offs between these techniques.

Figure 3

Vision Transformer architecture



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; 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, 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 [(Girshick, 2014; Ren, 2015)]; and finally YOLO, an object detection system that uses predicting systems in bounding boxes and class probabilities directly from the full image [(Redmon, 2017)]. Figure 2 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, and they are mixed up to synthesize patterns; this is generally referred to as token mixing. An example of a ViT architecture-based model is DETR (Detection Transformers); this workflow uses encoders and decoders to help understand images [(Carion, 2020)]. Figure 3 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, 2021)].

Interpretability Differences

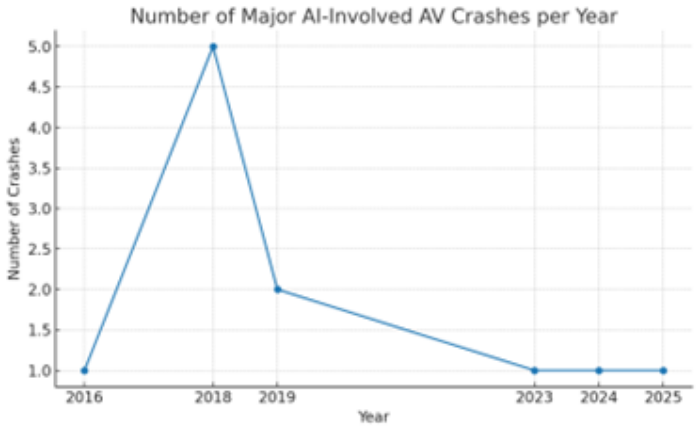
Inference patterns of different architectures differ under perturbations; this is explored and compared to provide a general overview of the interpretability. CNNs use Grad-CAM, a heatmap that reveals decision-driving regions in an image by tracing activations in the responses [(Selvaraju, 2017)]. This interpretation map allows development of a transparent model that can be trusted. Correspondingly, ViTs use self-attention maps to show 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; this is demonstrated by the consistent and accurate attention maps under the influence of noise [(Chefer, 2021; X. e. a. Mao, 2021; Zhou, 2022)]. While these interpretability differences are important, robustness metrics remain the primary lens of evaluation in this study.

Safety Risks in AI-Driven AVs

AI-based autonomous vehicle crash data is visualized in Figure 4 [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)]. The data, representing accidents from 2010 to 2025, shows a spike between 2016 and 2019, coinciding with the large-scale autonomous public testing [(California DMV, 2022; NTSB, 2020)]. The gradual decrease after regulations were synthesized resulted in a dynamic decline, leading to a stable and relatively low frequency of accidents. This is achieved by intensive testing in evaluation and training stages; additionally, computational simulations and metrics like mean average precision were used before public examinations. Risk factors like adverse weather and low-light contrast demonstrated instability in the model. Additionally, the

Figure 4

Line graph that illustrates the number of AI-involved AV crashes per year



fusion of different perspectives provided by sensors (LiDAR, radar, and camera inputs) [(Levinson, 2011)]. The first fatality recorded in a fully autonomous system [(NTSB, 2019)], an Uber ATG algorithm misclassified pedestrians' multiple times within 6 seconds. Resulting in an unstable path prediction, this caused the emergency brake system to disable due to a false-positive decision. A pedestrian was killed in Tempe, Arizona, due to the failure of the model classification techniques. It developed fear within the AV space. In the next sections in this paper, we will cover all factors and solutions to prevent fatalities along with lowering the graphs' accident frequency.

Types of Noise

Motion and Gaussian Blur

Blurs have the ability to hide or distort crucial information; the lack of clarity in vision can allow models to overlook key aspects that influence the interpretation. Swift movements can cause motion blur; Figure 5 provides a visual example from AIEase (n.d.). Free AI motion blur effect online. AIEase (an online blurring tool). These blurs decreases the interpretable information available for the model [(Nah, 2017)]. Identically Gaussian blurs are synthetic simulations of elements that can distract a model; these are often used in benchmark datasets to analyze

Figure 5

Motion and Gaussian Blur

situation-based noises [(Hendrycks & Dietterich, 2019)].

Fog and Haze

Fog adds scattered lighting, creating luminous areas that can wash out color and contrast in an image. The Figure 6 provides an idea of how much detail this perturbation can remove [(Henson, 2022)]. Detecting entities can become a big trouble under this type of noise; dehazing is a necessary concept to make data understandable [(Li, 2017)]. AOD-Net techniques provide a great network to prevent this type of error, allowing for quick and easy haze removal [(Li, 2017)].

Rain & Atmospheric Noise

As mentioned in the previous section, a very common and natural noise is rain. These create long, high-frequency streaks and varied lighting on artifacts. Figure 7 visualizes the noise [(LR-PS Tutorials, n.d.)]. Detailed restoration networks can mitigate these disturbances; cleansing techniques can preserve content while removing rain or any weather-based anomalies [(Fu, 2017)]. Rain-specific training allows models to ignore strokes in the image during the classification phase [(Zhang, 2019)].

Figure 6

Noise due to fog

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 8 provides a description and results when occlusions are present [(Ryu & Chung, 2021)]. This can prevent models from getting a global understanding of the context presented; it also contributes to object misclassification [(Hendrycks & Dietterich, 2019)]. 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)].

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 9 shows a low-contrast environment; the loss of clarity and detail is portrayed by the change in lighting [(Galer, n.d.)]. Texture-sensitive models are mainly affected by this noise [(Liu, 2017)]. To combat this, the addition of preprocessing methods like exposure corrections and dynamic range compensation is deemed necessary [(Chen, 2018)].

Figure 7

Blur caused by Rain

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 limitation heavily influences the model's interpretability. This section addresses solutions to solve these imperfections, methods like pixel-level cleaning, deblurring and visibility enhancements, weather-specific processing, and robust training. These approaches collectively improve clarity and preserve detailed information for answering prompts; additionally, they develop resilience against perturbations in images.

Pixel-Level Cleaning

Images tend to have different variations of light contrast; this can cause dynamic changes in interpretation as it makes it hard for models to split into layers and understand the global idea. The usage of histogram equalizations or CLAHE (Contrast Limited Adaptive Histogram Equalization) mitigates these risks. They spread out pixel intensity values to ensure darker regions become brighter and lighter regions become dimmer. This improves the global contrast of the picture, giving the model a genuine perspective of the image [(García, 2017; 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 when the model is able to compare sizes of objects to synthesize patterns, successfully adding to the

Figure 8

Examples depicting occlusion

(a) Occlusion by Preceding Vehicles (b) Occlusion by Shadow and Truncation

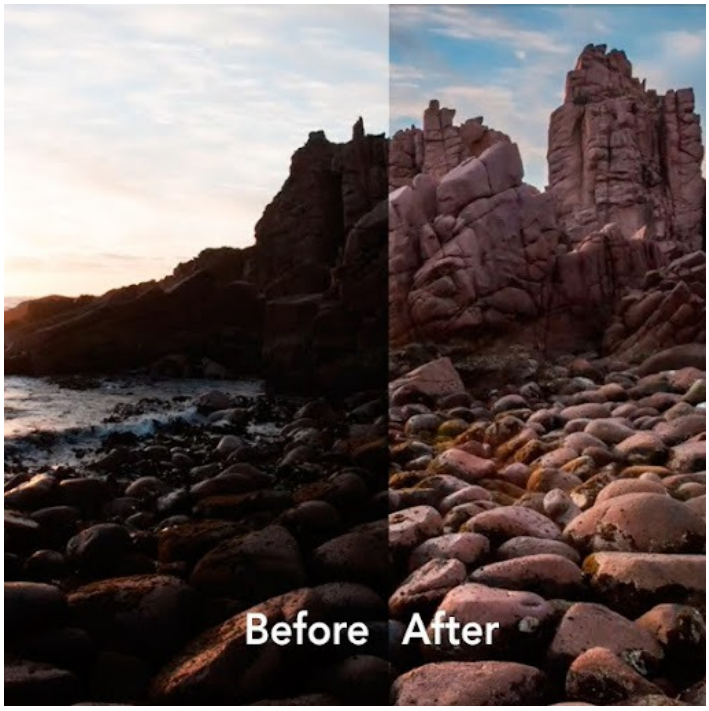
model's predictions and output [(Howard, 2017)].

Deblurring and Visibility Enhancement Methods

Blurs and visibilities have affected our human sights, causing inaccurate judgments and a lack of understanding of our surroundings. Figure 10 "What Is Imerge Pro?" (Manula, FXhome, 2021) 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 our eyes 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, 2017)]. All-in-One Dehazing (AOD) is a technique used to cleanse a picture from perturbations in a singular and contained space. Figure 11 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; it aids in the restoration of a sharp image [(Li, 2017)].

Figure 9

Difference in illumination



Weather-Specific and Rain Removal Pre-processing

Raindrops create streaks in images, acting as distractions and causing unwanted lines within a visual. This negatively impacts the model's stability; therefore, techniques like Detail-Recovery Network (DNN) eradicate it. DNNs use high-frequency isolations, where rain strokes usually appear, to remove these stripes. High-pass filters are applied to input images to carry out this procedure; continuous training on these specific perturbation models develops resilience toward it. Loss functions are integrated to balance rain removal and detail preservation; this can help preserve fine edges in a figure [(Fu, 2017)]. Other weather-based anomalies in an image are treated using weather-augmented training. It 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. [(Zhang, 2019)]

Figure 10

Motion and Gaussian Blur

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 robust 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, 2019). Although adding anomalies strengthens the model's interpretability, it fails to provide statistical evidence to fully address this concept's validity. A more developed yet naive approach involves weaving corrupted images and clean ones 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, 2019)].

Figure 11

AoD Motion and Gaussian Blur Removal

(a) Haze (b) AOD-Net

Model Strengths Across Perturbation Scenarios

The diversity in vision model architects allows 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, 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; also, they were able to adapt to partial occlusions. The strengths describe the following: Stability ensures reliable and balanced performance. Retention provides durable and strong memory. 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 trustability, and adaptability ensures versatile and evolving capacity.

Observation & Analysis

This section aims to provide a secondary analysis on CNNs' and ViT-based models' precision under the influence of natural or synthetic noise. 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

Table 1

Model strengths across different perturbation scenarios based on the literature

Perturbation Type	Scenario	Best Model	Strength
Environmental	Moderate rain	CNN	Stability
Environmental	Fog	CNN	Retention
Environmental	Low-light glare	ViT	Robustness
Natural	Snow	Hybrid CNN+ViT	Coverage
Natural	Dust	Hybrid CNN+ViT	Detection
Natural	Shadows	ViT	Resilience
Adversarial	Pixel noise	ViT	Confidence
Occlusion	Partial blockage	Hybrid CNN+ViT	Adaptability

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 below in Table 2. This subsection supports these claims by providing relevant statistics. To recap, CNN's main defect is its vulnerability to dynamic perturbation; they often fail under low contrast or high blur conditions due to reliance on local spatial features [(X. e. a. Mao, 2021)]. ViTs, on the other hand, perform better under disturbance due to global and self-attention mechanisms [(Zhou, 2022)]. The COCO benchmark dataset, includes a wide range of perturbations for CV tasks where Gaussian Blur could cause up to 45% mAP reduction (Mean Average Precision scores; they are used to compare and rate models, and their values range from 0 to 1 [(X. e. a. Mao, 2021)]) as compared to 25% for ViT architecture systems.

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. Cleaning can be done at different levels: architecture, dataset,

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, 2021)]
Mask R-CNN	CNN	Natural	63.1	49.4	13.7	[(Shankar, 2021)]
EfficientDet-D4	CNN	Natural	49.4	∼35	14.4	[(Zhou, 2022)]
DETR	ViT	Natural	42.0	∼32	10.0	[(C. e. a. Mao, 2023)]
Deformable DETR	ViT	Natural	45.4	∼34	11.4	[(C. e. a. Mao, 2023)]
DINO (Swin-L)	ViT	Natural	51.3	∼38	13.3	[(Zhou, 2022)]
Swin-L Detector	ViT	Natural	58.7	∼44	14.7	[(Zhou, 2022)]

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, 2015)]. This allows diagnosis of specific vulnerability areas in perception-based models.

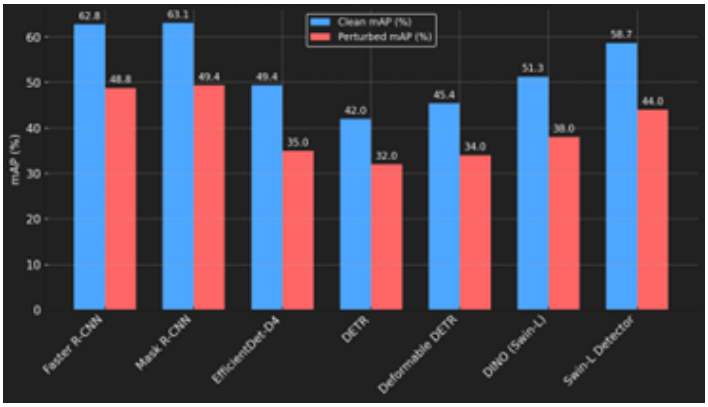
Key Observations & Analysis

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; the

Figure 12

Clean vs perturbed mAP under natural perturbations



drops vary 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.

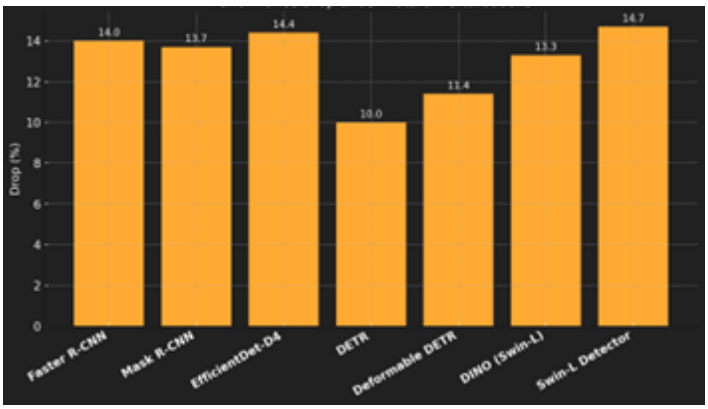
CNN Models: Robustness in Certain Scenarios

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; Table 2 supports this argument. EfficientDet-D4, regardless of 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 show 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.

Figure 13

Performance drop under natural perturbations



Higher Accuracy Does Not Guarantee Robustness

Although Swin-L detectors have a clean mAP of 58.7%, it falls to ∼44% under defects, this proves that a higher baseline precision does not refer to better perturbation resistivity. Similarly, another model with a relatable issue is EfficientDet-D4, where it holds a modest clean mAP yet suffers a comparable drop to the state-of-the-art CNNs and ViTs. This underlines 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 Tabel 2 corresponded to natural perturbations like motion blur and fog; this ensures 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.

Advanced Robustness Analysis

Traditional graphs (Figure 13, 12) illustrate clean vs perturbed mAP; this provides an understanding of the "average" performance. While these insights are helpful, these face-value

Table 3

Reliability distributions for CNNs versus ViTs.

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

perspectives hide critical weak points in the algorithm. A model can appear strong overall, yet crumble under corruptions in an image. It is an important constraint for AV tasks because the model deployment is delayed by rare but harmful failures [(Michaelis, 2019), (Wang et al., 2024)].

Multi-Metric Robustness Framework

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.

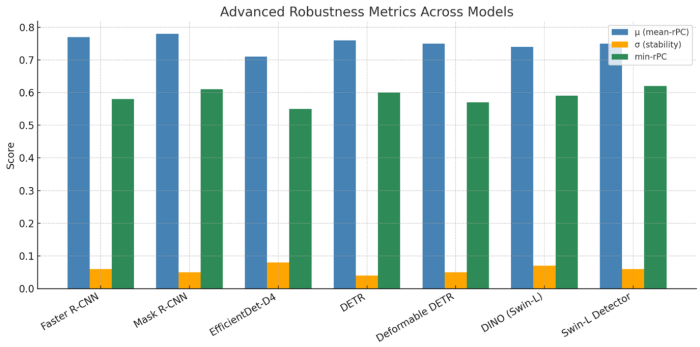
Metrics

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)].

$$\mu = \frac{1}{n}\sum_{i=1}^{N}\frac{mAP_i - mAP_{clean}}{mAP_{clean}} \tag{1}$$

Figure 14

Bar graph illustrating the advanced robustness metrics (μ, σ, min-rPC) for the selected models.



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 et al., 2025)].

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

min-rPC (worst-case) portrays 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 et al., 2021)].

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

Table 3 demonstrates reliability distributions for CNNs versus ViTs, highlighting that CNNs vary more widely between disturbances, while ViTs focus on stability, compromising on averages.

The bar graph in Figure 14 displays the three advanced parameters as mentioned above (μ, σ, min-rPC) for all 8 of the models. Figure 14 indicates CNNs' and ViTs' average performance, accuracy, and safety under noise. ViTs indicate trends, with lower σ metrics (0.04 for DETR); this shows greater accuracy. Although this is good, it simply shows a low min-rPC (0.60). Models like the Swin-L Detector have optimal worst-case robustness (0.62) and reasonable average robustness (0.75). CNNs, on the other hand, show greater μ (mean-rPC) on

average, fluctuating from 0.71 to 0.78, but algorithms like EfficientDet-D4 suffer from high volatility ($\sigma = 0.08$). DINO Swin-L performs consistently but does not beat the Swin-L Detector. The graph reveals that CNNs lead in average resilience, but ViTs provide safer lower limits. This is a crucial component that is vital for AV security.

Mask R-CNN shows a higher mean robustness ($\mu = 0.78$) and accuracy σ (0.05) when compared to Faster R-CNN and EfficientDet-D4. On a contrast basis, it outperforms Faster R-CNN in worst-case robustness by 0.03, this proves its supremacy in the design, balancing the accuracy and dependability, making 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.

Discussion

The comparative study of ViTs and CNNs provides insights into the frameworks' trade-offs that could potentially affect the algorithms' interpretability, robustness, and practicality in autonomous driving tasks. Both the architectures have significant benefits in different sectors of the domain. CNNs have greater potential to excel in local texture recognition; this allows them to achieve strong performance values under heavy detail-oriented and structured environments. Unfortunately, they struggle to maintain this under distortions that span the whole visual input; disturbances like fog and low light contrast are some examples. On the contrary, ViTs surpass CNNs in global understanding and self-attention. They provide reliable outputs when they are

under the influence of a large range of distortions. Yet the computational cost and latency factors drag the deployment of these architecture-based models. Additionally, interpretability creates another layer that separates these systems; CNNs are able to provide more intuitive Grad-CAM maps. This enables transparency and reveals influential regions in an image. However, ViTs are not able to match its opaque nature; the attention scores and segment-wise analysis make ViTs harder to understand, thereby reducing visibility in their decisions. The fundamental concept that allows these networks to make predictions is the dataset and the algorithm; they are the most significant components in a system. Therefore, benchmark datasets that exaggerate a model's stability must not be taken for analysis based on the face value. One main reason is that these databases cannot mimic the unpredictable real-world conditions. While the algorithms have enhanced significantly over the past years, hardware effectiveness becomes apparent as an understudied factor. Although ViTs have the capacity to outperform other architectures in robustness, CNNs still dominate in cost-effectiveness in resource-driven AV systems.

Future Steps

The next stage of development in the model must focus on integrating and adapting across algorithms rather than constraining the scope to isolated architecture comparison. There must be a dynamic switch from developing secluded systems to hybrid perception models that have the ability to adapt to diverse real-world noises. Coordinating various sensors like LiDAR and radar to simultaneously work with vision algorithms can enhance outputs; this eradicates the dependence on cameras. Although they have the capacity to illustrate the full picture, they are the most fragile when it comes to resilience in visual perturbation capture. The dynamic increase in the development of vision algorithms is influencing the domain significantly, but there are no standardized metrics to measure robustness accurately. To enable comparison, conventional measurements must be used to bring out the true potential of models. Current systems are trained on exaggerated perturbations to ensure maximum robustness. Yet this training method fails to

address the unpredictable and realistic conditions that humans come across every day. Establishing a real-world scenario-based database exposes the real-time performance and will allow models to find applicable patterns.

Conclusion

While CNNs and ViTs thrive in the field of autonomous vehicles, independently, they cannot provide absolutely reliable and consistent outputs. Even though CNNs offer full disclosure for decisions and accuracy, but they often suffer under various distortions in real-world scenarios. Faster R-CNN and Mask R-CNN lose over 13 to 14% of their accuracy when they face noise, as recorded in the paper, proving that they are unstable under noise. On the contrary, ViTs provide stability under perturbations; however, they lack interpretability and energy efficiency in real-world AV integrations. DETR shows the lowest drop in mAP score ($\sim$10%); additionally, Swin-L has the highest clean accuracy (58.7%), yet these advantages are dragged down due to its complex structure. Robustness should not be solely dependent on the model; the responsibility must be shared across the entire pipeline. Every component, like preprocessing and model framework, has an influence on the performance. Further analysis reveals that a hybrid pipeline that integrates both architectures can preserve performance under granular noise while maintaining accessibility. Therefore, they dominate single-architecture approaches in AV tasks. Ultimately models like EfficientDet-D4 only record a worst-case rPC of 0.55; this is the lowest score recorded by this paper. This rises as a pressing issue due to the rare but high-risk failure factor experienced by underperforming algorithms, making it a challenge to adapt models in real-world cases. The most promising path ahead is to adapt to hybrid architectures that include enhanced cross-sensor collaboration and methods to mitigate noise and evaluation benchmark datasets. This enables trustworthy, explainable, and resilient AV systems to attain a new perspective of the available architectures. Uniting these factors and architectures synthesizes a comprehensive and safe deployment of autonomous algorithms to strengthen driving tasks.

References

Badue, C. e. a. (2021). Self-driving cars: A survey. *Expert Systems with Applications*, *163*, 113791.

Bhojanapalli, S. e. a. (2021). Understanding robustness of transformers for image classification. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021*.

Bojarski, M. e. a. (2016). End-to-end learning for self-driving cars. *arXiv:1604.07316*.

Caesar, H. e. a. (2020). Nuscenes: A multimodal dataset for autonomous driving. *Computer Vision and Pattern Recognition (CVPR)*.

California DMV. (2022). Autonomous vehicle disengagement reports 2022. *Report*.

Carion, N. e. a. (2020). End-to-end object detection with transformers. *European Conference on Computer Vision (ECCV)*.

Chefer, H. e. a. (2020). Transformer interpretability beyond attention visualization. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Chefer, H. e. a. (2021). Transformer interpretability beyond attention visualization. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021)*.

Chen, Z. e. a. (2018). An attention-based deep learning framework for low-light image enhancement. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Cordts, M. e. a. (2016). The cityscapes dataset for semantic urban scene understanding. *Computer Vision and Pattern Recognition (CVPR)*.

Dai, J. e. a. (2017). Deformable convolutional networks. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*.

Dong, Z., Xu, X., He, S., Wu, Z., Xie, J., & Chen, T. (2025). Tire slip angle estimation based lateral stability control strategy for trajectory tracking scenarios of distributed drive

autonomous electric vehicles. *Control Engineering Practice*. https://doi.org/10.1016/j.conengprac.2025.106343

Dosovitskiy, A. e. a. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations (ICLR)*.

Fu, X. e. a. (2017). Removing rain streaks from a single image via deep detail network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Galer, M. (n.d.). Develop: Editing images with extreme contrast [free lightroom tutorial] [n.d.].

García, N. e. a. (2017).

Ghiasi, G. e. a. (2018). Dropblock: A regularization method for convolutional networks. *Advances in Neural Information Processing Systems (NeurIPS)*.

Girshick, R. e. a. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

He, H. e. a. (2024). Cf-cam: Cluster filter class activation mapping for reliable gradient-based interpretability. *Pattern Recognition*.

He, K. e. a. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hendrycks, D., & Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. *International Conference on Learning Representations (ICLR)*.

Henson, T. (2022, January). Before & after: Foggy morning on casco bay [[Photograph]. Todd Henson Photography].

Howard, A. e. a. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv:1704.04861*.

Janai, J. e. a. (2020). Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends in Computer Graphics and Vision*.

Kalra, N., & Paddock, S. (2016). Driving toward a wiser tomorrow. *RAND Corporation*.

Kang, S., & Seo, K. (2024). Sketch classification using vit. *JEET*, *19*, 4587–4593.

Khan, S. e. a. (2021). Transformers in vision: A survey. *ACM Computing Surveys*.

Lai-Dang, T. (2024). Interpretable medical imagery diagnosis with self-attentive transformers: A review of explainable ai for health care. *BioMedInformatics, 2024*.

Lambertenghi, G. e. a. (2025). Benchmarking image perturbations for testing automated driving assistance systems. *Proceedings of the IEEE International Conference on Software Testing, Verification and Validation (ICST 2025)*.

LeCun, Y. e. a. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2324.

Levinson, J. e. a. (2011). Towards fully autonomous driving: Systems and algorithms. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*.

Li, B. e. a. (2017). Aod-net: All-in-one dehazing network. *International Conference on Computer Vision (ICCV)*.

Liu, Y. e. a. (2017). Learning a deep single image brightening network with attention-based feature fusion. *IEEE International Conference on Image Processing (ICIP)*.

LR-PS Tutorials. (n.d.). Add a rain effect to the photo [n.d.].

Mao, C. e. a. (2023). Coco-o: A benchmark for object detectors under natural distribution shifts. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Mao, X. e. a. (2021). Adversarial attacks are reversible with natural supervision. *In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (pp. 641–651)*.

Michaelis, C. e. a. (2019). Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*.

Nah, S. e. a. (2017). Deep multi-scale convolutional neural network for image deblurring. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Naseer, M. e. a. (2021). Intriguing properties of vision transformers. *Advances in Neural Information Processing Systems (NeurIPS)*.

NTSB. (2019). Collision between vehicle controlled by developmental automated driving system and pedestrian, tempe, arizona, march 18, 2018. *Report*.

NTSB. (2020). Collision between vehicle controlled by developmental automated driving system and pedestrian, tempe, arizona, march 18, 2018. *Report*.

Paul, S., & Chen, Y. (2021). Vision transformers are robust learners. *In Proceedings of the AAAI Conference on Artificial Intelligence*.

Prakash, A. e. a. (2021). Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Redmon, J. e. a. (2017). Yolo9000: Better, faster, stronger. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ren, S. e. a. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems (NIPS)*.

Ryu, S.-E., & Chung, K.-Y. (2021). Appl. sci. 11(15), 7093. *Journal*.

Samek, W. e. a. (2015). Evaluating the visualization of what a deep neural network has learned. *International Conference on Learning Representations (ICLR) Workshop*.

Selvaraju, R. e. a. (2017). Grad-cam: Why did you say that? *International Conference on Computer Vision (ICCV)*.

Shankar, V. e. a. (2021). Do image classifiers generalize across time? *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 9661–9669.

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, *6*(1), 1–48.

Subbaswamy, A., Adams, R., & Saria, S. (2021). Evaluating model robustness and stability to dataset shift. *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*. https://proceedings.mlr.press/v130/subbaswamy21a.html

Szegedy, C. e. a. (2015). Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Thrun, S. (2010). Toward robotic cars. *Science*, *327*(5969), 1215–1215.

Vaswani, A. e. a. (2017). Attention is all you need. *Advances in Neural Information Processing Systems (NIPS)*.

Wang, S., Yu, Z., Jiang, X., Lan, S., Shi, M., Chang, N., Kautz, J., Li, Y., & Alvarez, J. M. (2024). Omnidrive: A holistic llm-agent framework for autonomous driving with 3d perception, reasoning and planning. *arXiv preprint arXiv:2405.01533*.

Yi, C., Yang, S., Li, H., Tan, Y.-P., & Kot, A. C. (2021). Benchmarking the robustness of spatial-temporal models against corruptions. *Advances in Neural Information Processing Systems (NeurIPS)*.

Yu, T. e. a. (2020). Bdd100k: A large-scale diverse driving video dataset. *Computer Vision and Pattern Recognition (CVPR)*.

Zhang, H. e. a. (2019). Image de-raining via a conditional generative adversarial network. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhou, W. e. a. (2022). Understanding the robustness in vision transformers. *arXiv:2201.03714*.

Zuiderveld, K. (1994). Contrast limited adaptive histogram equalization. *Graphics Gems IV*.

Review for **"Eyes on the Road: Elucidating ViTs and CNNs Under Real-World Noise"**

This paper reports the results of a thorough comparison between Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) during autonomous driving. It is a scientifically sound, thorough and well-written study on a timely research topic. A deep and thorough review of the literature is offered as strong background and context for the study. The application of well-established quantitative metrics to assess performance of these models when applied to publicly available datasets is informative and relevant. There is a focus on robustness and transparency which makes the results and discussion applicable to real-life scenarios. I consider this study suitable for publication provided that a few revisions are implemented first.

A) It is not fully clear where the results in a few of the figures come from. It appears that the authors have done some analysis on available data but this should be reported clearly and rigorously. Some more explanation on this (e.g. including an analysis methodology section) would be welcome. Also, the figures are good and rich but they could be integrated more with the main manuscript and the figure captions could be more detailed for clarity. Perhaps adding more structure in the paper to differentiate the content that is an outcome of literature review and the one that results from new analyses would enhance readability.
B) Some more insights on why CNNs are more interpretable than ViTs and how their combination would improve performance and robustness would be very informative here. Has this been implemented before or are there examples that corroborate this suggestion? What features of the algorithms underlie the differences that the author mentions and how can these be improved by the hybrid model? Ralting this interpretations back to the metrics that they used would make the point stronger and more supported.
C) A few sentences tend to be complicated with some level of repetition. Writing could be simplified and become tighter also in relevance to the improvements in structure and figure integration I mentioned above. This would enhance readability.

Overall this is a strong and novel research contribution on a relevant topic. The real-world approach and the thorough quantification can make the study impactful. The revisions suggested will enhance clarity and transparency in the final report.

My recommendation is

Accept with major revisions (acceptance conditional on satisfactory major revisions)

Overall comments

- Originality & Significance – This paper is very original! I liked the idea, and this is a very salient topic that the author engages with exhaustively
- Clarity & Structure – While the clarity in individual sections is very good, I feel like the paper could benefit from some shortening in the background sections to enhance the overall clarity. There was a lot of information presented, and at times, it didn't all connect to the main goals of the research.
- Use of Evidence & Research Methods – Overall, the evidence use was very thorough, especially in the literature review section. A major critique I have, however, is the lack of a methods section. It was unclear what the author was doing on their own and what was derived from the literature. The analysis section should be substantially expanded and motivated. Furthermore, connecting this section with the broader goals of the paper would really help comprehension
- Engagement with Literature – This was overall good. I think the engagement with the literature around the author's specific benchmarking analysis could be improved. For example, are their results in line with what the literature would dictate? What does this mean for the conclusions derived in the literature review section? I think better engagement with the literature would help the paper feel more cohesive; as of right now, it almost seems like the analysis was an afterthought
- Grammar & Language – The writing style is generally clear and coherent

Decision: Accept with major revisions

Detailed comments:

Abstract:

- Minor comment but if you use acronyms in your abstract (like mAP) you should define them, even if you do again later in your paper
- From the abstract, it is a little unclear whether you are writing a review paper or doing an analysis. In fact, I think that this is a little unclear in your entire paper. I'd make this crystal clear in your abstract

Introduction:

- I would add a specific "Introduction" header to make the transition from abstract clear
- You spend the beginning of the introduction talking about how autonomous vehicles are impacted by the data that they receive, and then transition into discussing the algorithms you will be reviewing. However, you never explicitly link that these algorithms are making the decisions for the car. I think it would be helpful to back up a little, and spend time talking about the flow of information in this system– first, the car records video information, then the video data is processed by algorithms (like CNNs), and based on the results of those algorithms, the car changes its behavior.
- You focus on cars that utilize image data – this is not exclusively used by all autonomous vehicles. In fact, many new cars (like you mention later) use lidar exactly as a response to the challenges of image disturbance. I think you should add a caveat earlier that improving image processing is one solution to improving car performance. This is okay, it's good scientific practice to be as specific about your conclusions as possible
- What is misclassification in this context? Even if it is obvious to you, it might not be for your reader (you should imagine young adults who aren't necessarily familiar with self-driving cars), so don't be afraid to be specific
- *"Understanding the way of predicting the accuracy is fundamental for humans to trust artificial intelligence systems in safety-critical sectors."*- this sentence is unclear to me. Maybe review for a typo
- Since your key gap is about interpretability, I would explain a little more how you think this information would be used. On one hand, you could argue that if a person is never reviewing the model outputs, it doesn't matter if the model is interpretable. Meanwhile, I agree that trust is crucial. Maybe to bolster your argument, give specific examples of how interpretability is important in building self-driving cars
- Explain more what the COCO benchmark set entails
- In general, the introduction lays out the research question and goal very clearly

Computer Vision and Autonomous Vehicles:

- Overall, I think this section is well reasoned and clear

- At times, I think your paper is a little lengthy. For example, the paragraph starting with *"This segment studies the use of Computer Vision...."* could be shortened to a one or two sentences
- I know you already have a lot of figures, but this section could be a good place for an overview schematic or graphical abstract showing the flow of information through the system. It's still a little unclear to the reader at where in the process of self-driving the algorithms that you are analyzing fit in
- The section *"How Modules Collaborate"* could be vastly reduced (or basically turned into the schematic I described earlier). In general, I don't think you need to tell me what you are discussing later in the paper, I think for holding the reader's attention, you should just get to the point succinctly
- In the *hardware* section, I don't think all cars use LiDAR sensors. Some use just cameras, like Teslas, famously. I would maybe be a more clear if you are focusing on specific types of cars
- This is the first time you use YOLO, please define or remove
- In the *Software* section, I would either define and explain *"Methods like HOG, SIFT, and Kalman filters"* or switch to a more general language. I don't think your audience will be familiar with these techniques and how they relate to the overall goals of understanding image disturbances. A good piece of advice I got in my scientific career is that if you don't want to explain something (or can't), don't include it in your paper/talk :)
- Again, everything in *"Synthesising an Intelligent System"* I think would be well served by a schematic
- *"After recent development in computer vision, new techniques like deep learning are much superior compared to their ancestors."* — this is a strong scientific claim, that I think you should have a citation for
- It'd be nice to have figure 1 closer to the point you are actually discussing (this is true for all the figures). Also, all your figures need more detailed captions. You should keep the title you have, but add an explanation of the image content. In general, a good rule is that the figure and its caption should be able to give enough information that someone could just look at the figures and understand their point
- You say an advantage of a CNN is robustness in adversarial attacks. What does this mean in the context of self-driving cars? If there isn't really relevance, maybe remove or cut down this section

Computer Vision Tasks within Autonomous Systems:

- I'm actually not sure the section "*Classical versus Deep Learning Approaches"* is needed. You introduce a lot of concepts you don't necessarily explain in detail, but I actually think it's not really required for your explicit goal of comparing the interpretability and performance of modern image processing algorithms. At this point, you've already introduced me to CNNs and ViT
- SImilarly, *"Relevance to interpretability"* contains a lot of information presented in your introduction that seems out of place

- Overall, I'd remove this section completely and just add the information to other appropriate sections. For example, consider adding the information about interpretability to *"Synthesising an Intelligent System"* as part of the software
- For figure 1, it's essential to tell me what the heatmaps actually mean when you reference this paper. The audience probably isn't going to be familiar with machine learning, so don't be afraid to feel like you are over explaining

Foundations of AI in Visual Perception Models :

- Why aren't there citations really in this section? Since I'm sure you're getting this information somewhere, you need to add citations
- *"Due to ViTs' recent growth, they are able to handle and provide significantly more accurate answers for complex scenes than the older CNN models."* - this is a very strong claim. Maybe it's not appropriate for this section of the paper where you are just explaining the overall architecture (and isn't evaluating this kind of the whole point of the paper?) Otherwise, needs a more thorough discussion and citation
- This section veers a little bit away from your main paper goals and just provides background information. I think because of that, the section could be condensed into one paragraph. Instead, I'd keep things tightly connected to your overall research goal as much as possible, or else the reader becomes a little fatigued
-

Structural Comparison: CNNs and ViTs:

- This section is overall nice and very important to your paper. As a reader, I kind of felt like "finally, we're getting to the point." It makes me wonder if you should save some of the information you have about CNNs and VIT in the *"Computer Vision and Autonomous Vehicles"* for this section and just shorten the earlier sections. For example, why not just have the figure 1 heatmap interpretability point here. It seems like a more appropriate place.
    - For example, the information on page 8 starting with "*Further research into this drawback has…* " would make more sense in this section to me since you are comparing performance
    - The *"Relevance to interpretability"* could also be condensed and put in the "*Interpretability Differences"* section. Maybe for the introductory material, just saying that it is important for models to be interpretable is enough
- You don't need the section starting with *"This section provides an overview"* . Maybe just one sentence to act as a transition

Safety Risks in AI-Driven AVs:

- Did you make figure 4 yourself? If so, I'd actually say you require a short methods section, because it is unclear where the data is from and how the number of crashes is calculated. It seems to me that the number is lower than I would expect, but maybe you

are focusing on accidents where the AI system was directly responsible for the crash. If so, a robust description of your data is needed
- Relatedly, are your results in figure 4 consistent with what other papers find on crash data? Contextualizing your results would help strengthen your point
- *"Risk factors like adverse weather and low-light contrast demonstrated instability in the model"* — were any of these crashes explicitly caused by these things? If so, state this
- To me, one crash a year is very safe and your point about one pedestrian fatality seems very very small in the context of car fatalities overall. It doesn't align with your conclusion that "*It developed fear within the AV space."*- instead I think you need citations of maybe how regulations were developed in response, or how it tangibly affected self-driving research.

Types of Noise and Common Techniques for Noise Mitigation:

- These sections are very good
- My only critique here is maybe to relate more directly your stated research focus on ViTs and CNNs. Maybe discussing how they would fit into the process of training and deploying these models. A diagram could be nice

Model Strengths Across Perturbation Scenarios:

- For this section, how are you deriving the information in table 1? Is it based on literature? If so, you should add citations to the table (and maybe a short description of how you extracted this information in a methods section)

Observation and Analysis:

- You need to explain the COCO dataset and what it entails
- Are you deploying models and doing empirical testing in this section? It is very unclear to me, so I think a more thorough explanation of what is the "secondary analysis" you are undertaking is warranted. If you are doing model experiments, you absolutely require a methods section. I would honestly cut a lot of the background if you are short on space and focus on explaining this, as it is absolutely crucial to your results
- Examples of things that need to be explained better:
    - How did you choose the models that are tested? At this point, this is the first I am hearing about things like Faster-R CNN or DETR. How does this relate to your CNN vs ViT focus? How do the models differ? I have no point of reference as the reader to evaluate your results. Even though you have some information about them in table 3, this is really late in your paper
    - Why mAP as the outcome?
    - How realistic are these perturbations?
    - What are the outcomes you are trying to predict? How realistic are these?
    - What image perturbations are you doing specifically? Are you doing just one type or multiple?

    - Do the models do any of the *Mitigation and Robustness Strategies* you carefully described earlier? This could provide context into your results for the reader (it wouldn't be that surprising that a model not trained on perturbed images would have reduced performance when tested on these types of images)
    - Are any of these models actually deployed in real self-driving cars? This would help to contextualize the magnitude of the results-is it really bad that the mAP decreases so much, or is this more of a research algorithm that isn't designed for deployment
- The advanced perturbation analysis is nice
- I'd include the equations in a methods section instead. You also need to define the parameters in the methods section and provide a caption for the equations
- Have any other papers used this benchmark? Are your results consistent with these papers? Why or why not? Engaging with the literature in this way could help add nuance to your results
- Table 3 states "*Reliability distributions for CNNs versus ViTs."* but you actually don't give me a lot of context for which algorithms are using CNN or ViT. I really like this section, but there are so many rich results, it loses the plot a little in respect to the stated goal in the introduction- explicitly comparing the VIT and CNN. Making this comparison more explicit in this table and in the whole section would really help your readers
- Overall, this is a cool section, but it felt a little secondary to other parts of the paper. I think you should be really explicit about how it connects to the literature that you spend so much time reviewing and in the conclusion section, why this analysis matter
-

Discussion:

- I'm kind of confused about the goal of this section, since you also have a conclusions section. To me, your whole paper is kind of a discussion. I expected this section to be more about the contextualization of your analysis results in the literature, so I was surprised to see it repeating some information you said earlier. Maybe you could combine the conclusions into one cohesive section that addresses the conclusions of your literature review and how your analysis results fit into the literature more broadly?
- You are missing citations throughout; it's okay to repeat citations you have already used, but things like "*CNNs have greater potential to excel in local texture recognition"* aren't directly interrogated by your analysis so need to have a citation

Future directions:

- I think also could benefit from some citations
- Since you did analysis, maybe you also want to state the future directions of analyses like yours?
- I honestly think this could be condensed into one discussion/conclusions section that goes in this paragraph order:
    - Overall results

- Contextualization of results in the literature
- Limitations and future directions
- Final thoughts and conclusions

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

[redacted]

Abstract

A misclassified citizen or an obstructed traffic light due to image degradation can lead to fatal consequences in autonomous driving systems. Such hindrances 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. Hence, 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 expose the regions of an image guiding decisions. Benchmark datasets such as Common Objects in Context (COCO) is 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: CNN, ViT, Self-attention, noise, perturbation, COCO, visualisation, Grad-CAM

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

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 it leads to dangerous misclassifications such as mistaking a pedestrian for a signpost or missing an obstructed traffic light. If they fail to make accurate decisions, it will 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, 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, 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 way of predicting 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 the influential regions in an image [(Selvaraju, 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 affects 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 their own 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? Therefore, 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 as mentioned above.

This paper conducts a secondary analysis and a literature review of prior work on CNN and ViT robustness under noise. Observations concluded are focused 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. It 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, 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 deployment of models in this safety-critical field. It also raises attention to 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 enable such environments to become 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.

Background and Literature Review

Computer Vision and Autonomous Vehicles

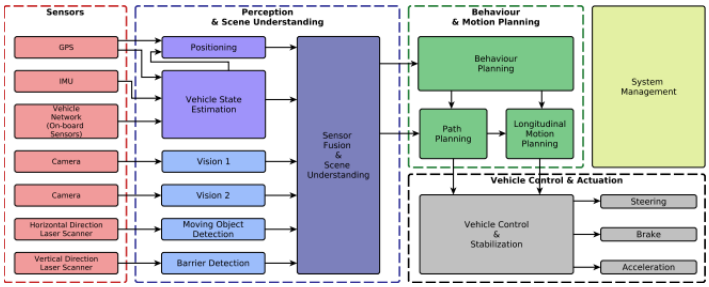
This section examines how Computer Vision (CV) helps autonomous driving systems process raw sensor data into meaningful information, enabling machines to interpret visuals like humans. It highlights how CV models adapt to environmental changes, strengthen feature extraction, and improve safety in real-world 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/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

Figure 1

This figure shows how information moves through an autonomous driving system. 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)].



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; 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.

Behaviour 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, 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 movement of vehicles to adapt to its environment; these estimations are directly correlated to the sensory inputs and the models' ability to interpret. 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)].

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. LiDAR sensors (Light Detection and Ranging), which are used for capturing depth in the pictures, radars for motion detection and cameras for their enhanced ability to capture high-resolution images. These sensors benefit the object detection and semantic vision for the models [(Levinson, 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, 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 development in computer vision, new techniques like deep learning are much superior compared to their ancestors [(Khan, 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 the 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. e. a. Mao, 2021)][(Zhou, 2022)]. The recent evolution has brought the limelight to spread over computer vision; this motivates researchers to identify more robust and interpretable models for use within the AV field [(H. e. a. He, 2024)][(Samek, 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 , 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, 2017)].

Foundations of AI in Visual Perception Models

This section exposes 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 identifies differences between the model's output values, it provides feedback on the precision of the system such that lower loss is higher performance [(Goodfellow et al., 2016)]. These steps allows 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, 2021)][(Z. e. a. Liu, 2021)].

Structural Comparison: CNNs and ViTs

This section 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 solve a common problem expose different perspectives and solutions, causing the dynamic trade-offs between these techniques. Some architectures shows 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 extrapolation 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 colour 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, 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. e. a. He, 2024)].

Figure 2

Figure Comparison between CNN and ViT heat maps under perturbed visuals. 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)]

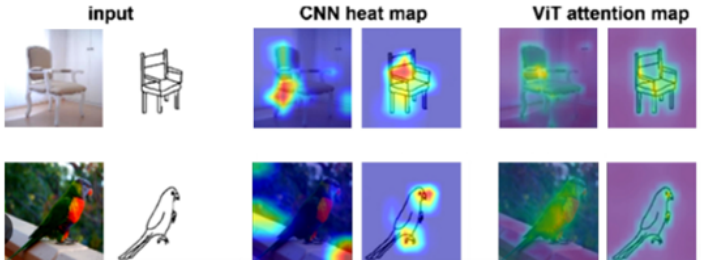
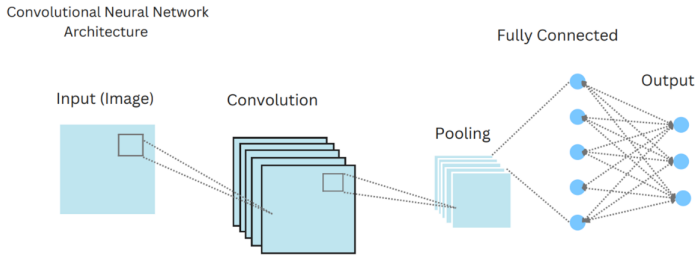


Figure 3

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

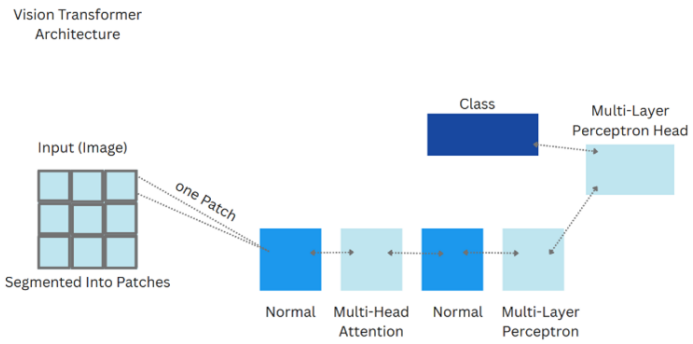


Feature Processing

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; 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

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.



needs of embedded systems, making them an ideal candidate for embedded system deployments [(Szegedy, 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, 2015)]; and finally YOLO, an object detection system that uses predicting systems in bounding boxes and class probabilities directly from the full image [(Redmon, 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, and they are mixed up to synthesize patterns; this is generally referred to as token mixing. An example of a ViT architecture-based model is DETR (Detection Transformers); this workflow 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, 2017)]. This interpretation map allows development of a transparent model that can be trusted. Correspondingly, ViTs use self-attention maps to show 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; this is demonstrated by the consistent and accurate attention maps under the influence of noise [(Chefer, 2021)][(X. e. a. Mao, 2021)][(Zhou, 2022)]. While these interpretability differences are important, robustness metrics remain the primary lens of evaluation in this study.

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 is a domain that will be positively affected as transparency in an architecture becomes more abundant [(Samek, 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 colour 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, 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

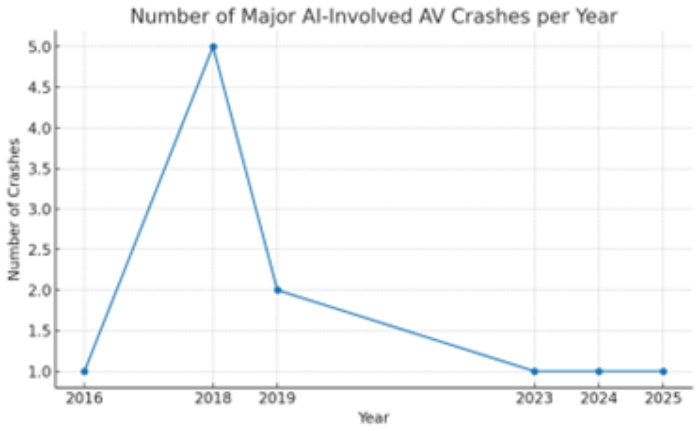
models that focus on relevant visual elements [(H. e. a. He, 2024)].

Safety Risks in AI-Driven AVs

AI-based autonomous vehicle crash data is visualized in Figure 5 [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)]. The data was collected from public regulatory and investigation reports or example 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.

Figure 5

Line graph that illustrates the number of AI-involved AV crashes per year



The data, presents accidents from 2010 - 2025, shows an increase between 2016 - 2019, due to the large-scale autonomous testing [(DMV, 2022)][(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 a reduction 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) [(Levinson, 2011)] strengthened object detection reliability, though limitations still existed. The first fatality recorded in a fully autonomous system [(NTSB, 2019)] occurred when the Uber ATG algorithm misclassified a pedestrian multiple times within six seconds. 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. It developed fear and caution within the AV space. In the next sections of this paper, we will cover all factors and solutions to prevent fatalities along with lowering the graph's accident frequency.

Types of Noise

During training noise is intentionally injected in 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, 2017)][(Krizhevsky et al., 2012)]. On the other hand, ViTs use patch-level normalization and adaptive positional encoding to work around the road block [(Dosovitskiy, 2020)][(Touvron, 2021)]. When it comes to deployment Sensor fusion helps reduce the impact of noise in

perception modules [(Levinson, 2011)][(Chen, 2018)]. Preprocessing will remove environmental distortions ensuring that visual inputs are consistent, this is crucial as both ViTs and CNNs depend on clean images.

Figure 6

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

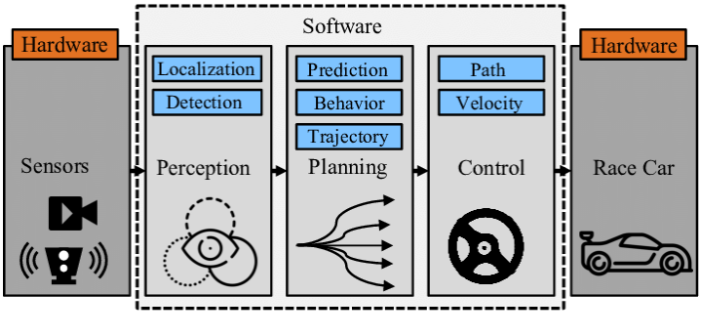


Figure 7

Shows how motion and Gaussian blur distort a picture. Fast movement or synthetic blur hides important details and confuses the model when detecting objects.

Motion and Gaussian Blur

Blurs have the ability to hide or distort crucial information; the lack of clarity in vision can allow models to overlook key aspects that influence the interpretation. Swift movements can cause motion blur; Figure 7 provides a visual example from (AIEase (n.d.). Free AI motion blur effect online. AIEase (an online blurring tool)). These blurs decreases the interpretable information available for the model [(Nah, 2017)]. Identically Gaussian blurs are synthetic simulations of elements that can distract a model; these are often used in benchmark datasets to analyze situation-based noises [(Hendrycks & Dietterich, 2019)].

Figure 8

Shows how fog removes color and contrast, washing out the picture. The model loses small details and can miss objects on the road.

Fog and Haze

Fog adds scattered lighting, creating luminous areas that can wash out color and contrast in an image. The Figure 8 provides an idea of how much detail this perturbation can remove [(Henson, 2022)]. Detecting entities can become a big trouble under this type of noise; dehazing is a necessary concept to make data predictable [(Li, 2017)]. AOD-Net techniques provide a great network to prevent this type of error, allowing for quick and easy haze removal [(Li, 2017)].

Figure 9

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

Rain & Atmospheric Noise

As mentioned in the previous section, a very common and natural noise is rain. These create long, high-frequency streaks and varied lighting on artifacts. Figure 9 visualizes the noise [(**lr-ps2017**)]. Detailed restoration networks can mitigate these disturbances; cleansing techniques can preserve content while removing rain or any weather-based anomalies [(Fu, 2017)]. Rain-specific training allows models to ignore strokes in the image during the classification phase [(Zhang, 2019)].

Figure 10

Example of how blocking objects hide what's behind them. It shows how occlusion can confuse the model by cutting off parts of the image.

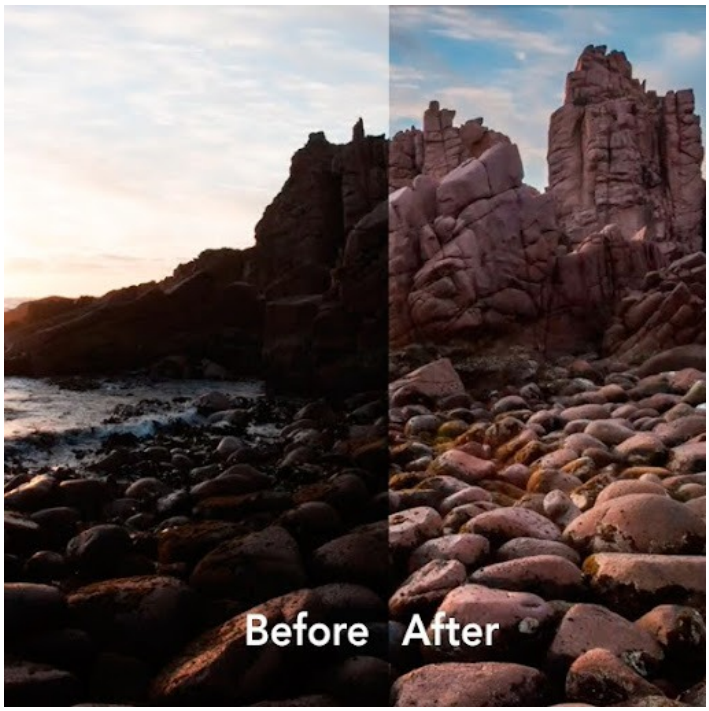
(a) Occlusion by Preceding Vehicles (b) Occlusion by Shadow and Truncation

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 global understanding of the context presented; it also contributes to object misclassification [(Hendrycks & Dietterich, 2019)]. 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 11

Shows how lighting changes between bright and dark areas confuse the model. Loss of texture and contrast reduces what it can detect.



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; the loss of clarity and detail is portrayed by the change in lighting [(**edwards2025**)]. Texture-sensitive models are mainly affected by this noise [(Y. e. a. Liu, 2017)]. To combat this, the addition of preprocessing methods like exposure corrections and dynamic range compensation is deemed necessary [(Chen, 2018)].

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 limitation heavily influences the model's interpretability. This section addresses solutions to solve these imperfections, methods like pixel-level cleaning, deblurring and visibility enhancements, weather-specific processing, and robust training mitigate these perturbations. These approaches collectively improve clarity and preserve detailed information for answering prompts; additionally, they develop resilience against perturbations in images.

Pixel-Level Cleaning

Images tend to have different variations of light contrast; this can cause dynamic changes in interpretation as it makes it hard for models to split into layers and understand the global idea. The usage of histogram equalizations or CLAHE (Contrast Limited Adaptive Histogram Equalization) mitigates these risks. They spread out pixel intensity values to ensure darker regions become brighter and lighter regions become dimmer. This improves the global contrast of the picture, giving the model a 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 compare sizes of objects to

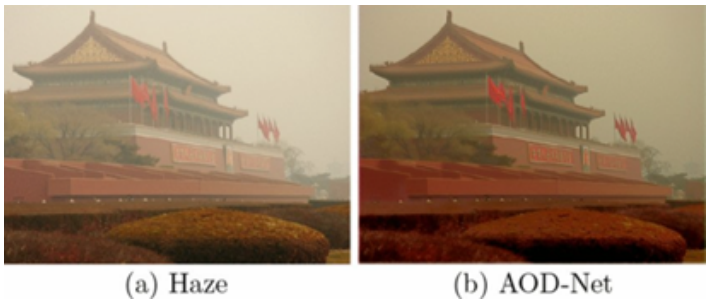
synthesize patterns, successfully adding to the model's predictions and output [(Howard, 2017)].

Figure 12

Motion and Gaussian Blur

Figure 13

AoD Motion and Gaussian Blur Removal



(a) Haze (b) AOD-Net

Deblurring and Visibility Enhancement Methods

Blurs and visibilities have affected human sights, causing inaccurate judgments and a lack of understanding of the surroundings. Figure 12 ("What Is Imerge Pro?" Manula, FXhome, 2021) 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 our eyes 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, 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; it aids in the restoration of a sharp image [(Li, 2017)].

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 its significance and eradicate the effect. DNNs use high-frequency isolations, where rain strokes usually appear, to remove these stripes. High-pass filters are applied to input images to carry out this procedure; continuous training on these specific perturbation models develops resilience toward it. Loss functions are integrated to balance rain removal and detail preservation; this can help preserve fine edges in a figure [(Fu, 2017)]. Other weather-based anomalies in an image are treated using weather-augmented training. It 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. [(Zhang, 2019)]

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 robust 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, 2019). Although adding anomalies strengthens the model's interpretability, it fails to provide statistical evidence to fully address this concept's validity. A more developed yet naive approach involves weaving corrupted images and clean ones 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, 2019)].

Table 1

Model strengths across different perturbation scenarios based on the literature

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

Model Strengths Across Perturbation Scenarios

The diversity in vision model architecture allows 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, 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; also, they were able to adapt to partial occlusions. The strengths describe the following: stability ensures reliable and balanced performance, retention provides durable and strong memory. 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 trustability, and adaptability ensures versatile and evolving capacity.

Methods

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 the broader and more reliable view, since the values are taken from peer-reviewed empirical papers rather than single controlled runs.

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 (), stability (), and minimum robustness coefficient (min-rPC). They were derived using the following formulations:

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, 2021)].

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

(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)].

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

min-rPC (worst-case) portrays 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, 2021)].

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

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

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

Collecting clean and corrupted mAP(mean average precision) results from the benchmark datasets.

Normalizing results across the architectures for consistent comparison.

Computing , , and min-rPC for each natural perturbation type.

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 the models used in this study follows 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, 2021; Zhou, 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; Z. e. a. Liu, 2021; C. e. a. Mao, 2023)].

CNNs pull out features via sequential convolution and pooling, while RPN layers separate

objects and BiFPN combines features across scales [(Michaelis, 2019; Wang, 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 [(Z. e. a. Liu, 2021; Zhou, 2022)]. The models were tested on COCO with comparable noise conditions [(Hendrycks & Dietterich, 2019; Lin, 2014)] , in addition to robustness was calculated using mAP, , , 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, 2017; Nah, 2017; Selvaraju, 2017)].

The studies used in this secondary analysis were filtered through a structured literature screening process. Searches were done using Google Scholar, IEEE Xplore, arXiv, and SpringerLink using 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. Papers that used standardized benchmarks such as COCO or ImageNet-C were prioratized and clearly specified noise types and evaluation metrics. This ensured methodological consistency across compared results and minimized dataset or reporting bias.

Observation & Analysis

This section aims to provide a secondary analysis on CNNs’ 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 diferent 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, 2014; C. e. a. Mao, 2023)]. It is the most used dataset for evaluating object segmentation and detection, additionally it forms the basis for all mAP (mean average precision) 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, 2019; C. e. a. Mao, 2023)] covering both environmental and synthetic distortions typically encountered in road scenes.

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, 2021)]
Mask R-CNN	CNN	Natural	63.1	49.4	13.7	[(Shankar, 2021)]
EfficientDet-D4	CNN	Natural	49.4	∼35	14.4	[(Zhou, 2022)]
DETR	ViT	Natural	42.0	∼32	10.0	[(C. e. a. Mao, 2023)]
Deformable DETR	ViT	Natural	45.4	∼34	11.4	[(C. e. a. Mao, 2023)]
DINO (Swin-L)	ViT	Natural	51.3	∼38	13.3	[(Zhou, 2022)]
Swin-L Detector	ViT	Natural	58.7	∼44	14.7	[(Zhou, 2022)]

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 below in Table 2. The models selected for comparison are: Faster R-CNN, Mask R-CNN, EfficientDet-D4, DETR, Deformable DETR, DINO, and Swin-L. They were chosen based on their leading benchmark architectures for object detection [(Carion, 2020; Ren, 2015; Zhou, 2022)] . Each model captures a different design: region-based proposals

(CNNs) against transformer-based global attention (ViTs). They enable a balanced cross-architecture analysis.

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, allowing both detection accuracy and completeness [(Hendrycks & Dietterich, 2019; C. e. a. Mao, 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, 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, 2021; Zhou, 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 they 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, 2011)].

Figure 14

Graph comparing model accuracy on clean images versus noisy ones. Every model drops in performance when exposed to natural distortions

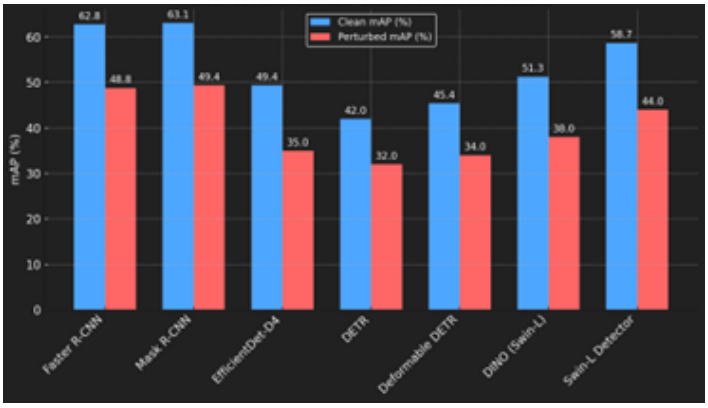
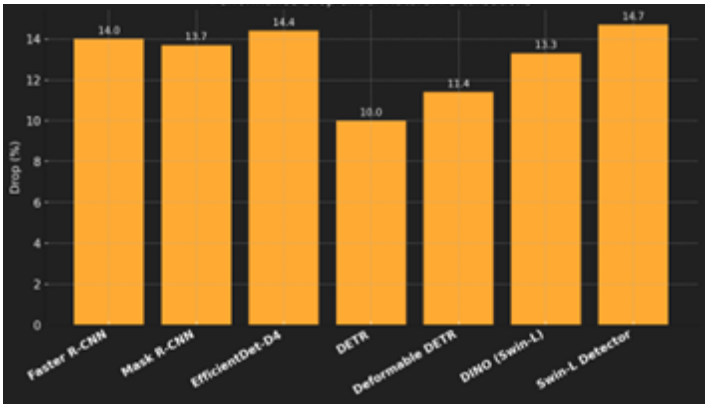


Figure 15

Shows how much each model's accuracy falls because of noise. ViTs lose less accuracy than CNNs, proving stronger stability.



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; the drops vary 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; Table 2 supports this argument. EfficientDet-D4, regardless of 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 show 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, this proves that a higher baseline precision does not refer to better perturbation resistivity. Similarly, another model with a relatable issue is EfficientDet-D4, where it holds a modest clean mAP yet suffers a comparable drop to the state-of-the-art CNNs and ViTs. This underlines 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 Tabel 2 corresponded to natural perturbations like motion blur and fog; this ensures 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.

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

Advanced Robustness Analysis

Traditional graphs (Figure 15,14) illustrate clean vs perturbed mAP; this provides 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

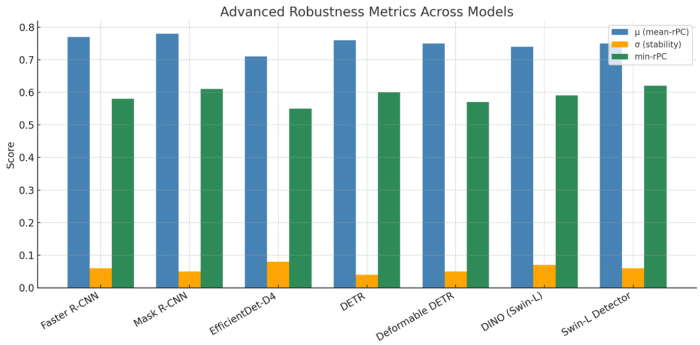
crumble under corruptions in an image. It is an important constraint for AV tasks because the model deployment is delayed by rare but harmful failures [(Michaelis, 2019), (Wang, 2024)].

Multi-Metric Robustness Framework

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.

Figure 16

graph comparing three robustness metrics for all models. CNNs score higher on average strength, while ViTs show more stability and better worst-case results



The bar graph in Figure 16 displays the three advanced parameters as mentioned above (μ, σ, min-rPC) for all 8 of the models. Figure 16 indicates CNNs' and ViTs' average performance, accuracy, and safety under noise. ViTs indicate trends, with lower σ metrics (0.04 for DETR); this shows greater accuracy. Although this is good, it simply shows a low min-rPC (0.60). Models like the Swin-L Detector have optimal worst-case robustness (0.62) and reasonable average robustness (0.75). CNNs, on the other hand, show greater μ (mean-rPC) on average, fluctuating from 0.71 to 0.78, but algorithms like EfficientDet-D4 suffer from high volatility ($\sigma = 0.08$). DINO Swin-L performs consistently but does not beat the Swin-L Detector. The graph reveals that CNNs lead in average resilience, but ViTs provide safer lower limits. This

is a crucial component that is vital for AV security.

Mask R-CNN shows a higher mean robustness ($\mu = 0.78$) and accuracy σ (0.05) when compared to Faster R-CNN and EfficientDet-D4. On a contrast basis, it outperforms Faster R-CNN in worst-case robustness by 0.03, this proves its supremacy in the design, balancing the accuracy and dependability, making 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.

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. e. a. Mao, 2021)].

On the other hand, ViTs are built to understand images globally through self-attention mechanisms [(Zhou, 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, 2015)]. CNNs build more interpretable Grad-CAM heatmaps that make it clear where the model is focusing, whereas 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, 2014)] also becomes a concern because they do not fully reflect unpredictable conditions faced in the real world [(Michaelis, 2019; Wang, 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.

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, 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 and hide the 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 percent mAP drop for DETR and 13 to 14 percent 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.

References

Badue, C. e. a. (2021). Self-driving cars: A survey. *Expert Systems with Applications*.

Bhojanapalli, S. e. a. (2021). Understanding robustness of transformers for image classification. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Bojarski, M. e. a. (2016). End-to-end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*.

Caesar, H. e. a. (2020). Nuscenes: A multimodal dataset for autonomous driving. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Carion, N. e. a. (2020). End-to-end object detection with transformers. *European Conference on Computer Vision (ECCV)*.

Chefer, H. e. a. (2020). Transformer interpretability beyond attention visualization. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Chefer, H. e. a. (2021). Transformer interpretability beyond attention visualization. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Chen, Z. e. a. (2018). An attention-based deep learning framework for low-light image enhancement. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Cordts, M. e. a. (2016). The cityscapes dataset for semantic urban scene understanding. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Dai, J. e. a. (2017). Deformable convolutional networks. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*.

DMV, C. (2022). Autonomous vehicle disengagement reports 2022. *Report*.

Dong, Z. e. a. (2025). Tire slip angle estimation based lateral stability control strategy for trajectory tracking scenarios of distributed drive autonomous electric vehicles. *Control Engineering Practice*.

Dosovitskiy, A. e. a. (2020). An image is worth 16×16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Fu, X. e. a. (2017). Removing rain streaks from a single image via deep detail network. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ghiasi, G. e. a. (2018). Dropblock: A regularization method for convolutional networks. *Advances in Neural Information Processing Systems (NeurIPS)*.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. *MIT Press*.

He, H. e. a. (2024). Cf-cam: Cluster filter class activation mapping for reliable gradient-based interpretability. *Pattern Recognition*.

He, K. e. a. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hendrycks, D., & Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. *International Conference on Learning Representations (ICLR)*.

Henson, T. (2022). Before and after: Foggy morning on casco bay. *Todd Henson Photography*.

Howard, A. e. a. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Janai, J. e. a. (2020). Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends in Computer Graphics and Vision*.

Kalra, N., & Paddock, S. (2016). Driving toward a wiser tomorrow. *RAND Corporation*.

Kang, S., & Seo, K. (2024). Sketch classification using vit. *JEET*.

Khan, S. e. a. (2021). Transformers in vision: A survey. *ACM Computing Surveys*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*.

Lai-Dang, T. (2024). Interpretable medical imagery diagnosis with self-attentive transformers: A review of explainable ai for health care. *BioMedInformatics*.

Lambertenghi, G. e. a. (2025). Benchmarking image perturbations for testing automated driving assistance systems. *Proceedings of the IEEE International Conference on Software Testing, Verification and Validation (ICST)*.

LeCun, Y. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*.

Levinson, J. e. a. (2011). Towards fully autonomous driving: Systems and algorithms. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*.

Li, B. e. a. (2017). Aod-net: All-in-one dehazing network. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Lin, T. e. a. (2014). Microsoft coco: Common objects in context. *European Conference on Computer Vision (ECCV)*.

Liu, Y. e. a. (2017). Learning a deep single image brightening network with attention-based feature fusion. *IEEE International Conference on Image Processing (ICIP)*.

Liu, Z. e. a. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Mao, C. e. a. (2023). Coco-o: A benchmark for object detectors under natural distribution shifts. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Mao, X. e. a. (2021). Adversarial attacks are reversible with natural supervision. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Michaelis, C. e. a. (2019). Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*.

Nah, S. e. a. (2017). Deep multi-scale convolutional neural network for image deblurring. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

NTSB. (2019). Collision between vehicle controlled by developmental automated driving system and pedestrian, tempe, arizona, march 18, 2018. *Report*.

NTSB. (2020). Collision between vehicle controlled by developmental automated driving system and pedestrian, tempe, arizona, march 18, 2018. *Report*.

Paul, S., & Chen, Y. (2021). Vision transformers are robust learners. *Proceedings of the AAAI Conference on Artificial Intelligence*.

Prakash, A. e. a. (2021). Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Redmon, J. e. a. (2017). Yolo9000: Better, faster, stronger. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ren, S. e. a. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems (NeurIPS)*.

ResearchGate. (2025). Autonomous vehicles on the edge: A survey on autonomous vehicle racing – scientific figure on researchgate. *ResearchGate*.

Ryu, S.-E., & Chung, K.-Y. (2021). Occlusion-aware object detection and classification in autonomous systems. *Applied Sciences*.

Samek, W. e. a. (2015). Evaluating the visualization of what a deep neural network has learned. *International Conference on Learning Representations (ICLR) Workshop*.

Selvaraju, R. e. a. (2017). Grad-cam: Why did you say that? *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Shankar, V. e. a. (2021). Do image classifiers generalize across time? *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Shorten, C., & Khoshgoftaar, T. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*.

Subbaswamy, A. e. a. (2021). Evaluating model robustness and stability to dataset shift. *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*.

Szegedy, C. e. a. (2015). Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Taş, Ö., Kuhnt, F., Zöllner, J. M., & Stiller, C. (2016). Functional system architectures towards fully automated driving. *IEEE Intelligent Vehicles Symposium (IV)*.

Touvron, H. e. a. (2021). Training data-efficient image transformers and distillation through attention. *Proceedings of the International Conference on Machine Learning (ICML)*.

Vaswani, A. e. a. (2017). Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*.

Wang, S. e. a. (2024). Omnidrive: A holistic llm-agent framework for autonomous driving with 3d perception, reasoning and planning. *arXiv preprint arXiv:2405.01533*.

Yi, C. e. a. (2021). Benchmarking the robustness of spatial-temporal models against corruptions. *Advances in Neural Information Processing Systems (NeurIPS)*.

Yu, T. e. a. (2020). Bdd100k: A large-scale diverse driving video dataset. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, H. e. a. (2019). Image de-raining via a conditional generative adversarial network. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhong, Z. e. a. (2017). Random erasing data augmentation. *arXiv preprint arXiv:1708.04896*.

Zhou, W. e. a. (2022). Understanding the robustness in vision transformers. *arXiv preprint arXiv:2201.03714*.

Zuiderveld, K. (1994). Contrast limited adaptive histogram equalization. *Graphics Gems IV*.

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

[redacted]

Abstract

~~A misclassified citizen or an obstructed traffic light due to image degradation can lead to fatal consequences in autonomous driving systems. Such hindrances 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. Hence, this paper explores how convolutional neural networks and vision transformers respond to altered visual inputs in autonomous scenarios. To understand a model's focus, we propose analyzing the attention or focus of a model under the influence of perturbed images. Additionally, we investigate the impacts of noise cleaning techniques on the model and its stability by using metrics like mAP. Robustness scores forms the core for evaluating reliability in this paper, while interpretability methods such as Grad-CAM and attention maps are complementary tools that reveal guiding regions in an image that influence decisions. Utilizing COCO, a standardized benchmark dataset, enables a fair study of these models. This approach will reveal the best models in each architecture after careful comparison using the above methods.~~

A misclassified citizen or an obstructed traffic light due to image degradation can lead to fatal consequences in autonomous driving systems. Such hindrances 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. Hence, 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 expose the regions of an image guiding decisions. Benchmark datasets such as Common Objects in Context (COCO) is 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.

Formatted: Font: 12 pt

Formatted: Right: 0.49", Space Before: 0.05 pt, Line spacing: Multiple 1.73 li

Keywords: CNN, ViT, Self-attention, noise, perturbation, ~~nuScenes~~COCO, visualisation, Grad-CAM

Formatted: Right: 1.06"

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

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 it leads to dangerous misclassifications such as mistaking a pedestrian for a signpost or missing an obstructed traffic light. If they fail to make accurate decisions, it will 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, 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. ~~Real-world noise ranges from motion blurs to sensor noises which can be caused by swift movement during the exposure time. Natural noises like fog, rain, and blurs are common types of perturbation that autonomous driving models face daily.~~ 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, 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 way of predicting ~~the~~ accuracy is fundamental for humans to trust artificial intelligence systems in safety-critical sectors. Interpretability is important because it

Formatted: Font: 12 pt

Formatted: Font: 12 pt

Formatted: Font: 12 pt

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 the influential regions in an image [(Selvaraju, 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 affects 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 their own strengths but are not yet perfect.

Formatted: Space Before: 0.25 pt

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? Therefore, 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 as mentioned above.

This paper conducts a secondary analysis and a literature review of prior work on CNN and ViT robustness under noise. Observations concluded are focused 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. It allows a fair and consistent comparison of models under the same conditions. ~~The analysis process involves visualizing key findings through graphs for fair evaluations. Visual data used in prior studies will go through comparisons to obtain usable figures and trends in models. The referenced data that these models will be tested on are COCO benchmark datasets; occasionally real-world noise is used too.~~ Stability of the models under differing levels of perturbations allows simulations of artificial environments [(Caesar, 2020;

Yu, 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 deployment of models in this safety-critical field. It also raises attention to 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 enable such environments to become 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.

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)].

~~This segment studies the use of Computer Vision (CV) in perspective-driven autonomous driving systems; the model typically converts raw sensor inputs into structured and valuable data. Computer vision allows machines to understand and interpret visual data; it enables computation systems to emulate the human visual system through advanced technologies. CV models enable robust object detection and environmental understanding, resulting in better decision-making support under perturbations. Additionally, CV models are trained to adapt to surroundings like changes in the weather and lighting, providing more powerful feature extraction abilities. Thereby improving safety in navigation tasks in real-world descriptions [(Cordts, 2016; Janai, 2020)]. This section explores the various components that a synthetic system can control in the modernized transport systems. Furthermore, it delves into the link between the software and hardware components within this autonomous pipeline.~~

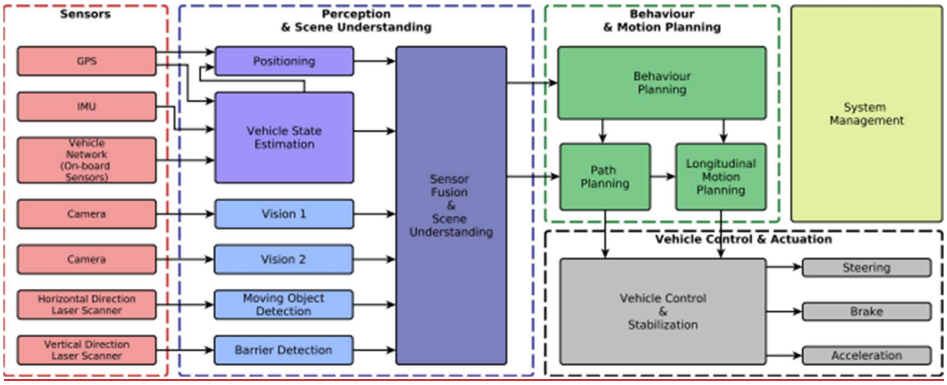
Formatted: Right: 0.49"

Controlling Factors

Core mechanisms that enable the vehicle to drive are controlled by the autonomous system. ~~These components include steering, braking, signaling, indicators, honking, lighting, and throttle [(Thrun, 2010)]. These mechanisms are commanded using the decisions made by the system.~~ The decision algorithm is heavily influenced by the objects/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; 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.

Figure 1

This figure shows how information moves through an autonomous driving system. 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, Kuhnt, Zöllner, & Stiller, 2016)].



Behaviour 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, 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 movement of vehicles to adapt to its environment; these estimations are directly correlated to the sensory inputs and the models' ability to interpret. Industrial applications of these comprehensive models decompose the problem into steps and run them through a pipeline of layers. These layers integrate this information in forecasts and forethoughts of the outcomes. NVIDIA Drive and Waymo have integrated modular pipelines in their autonomous vehicle systems [(Bojarski, 2016)].

~~How Modules Collaborate~~

~~cThis section outlines the sensing and perception pipelines that underscore autonomous driving systems, exploring how sensor data is processed through recognition layers to make real-time decisions. The suggested pipeline consists of LiDAR, radar, and camera sensors for the visual inputs, as this is connected to the object detection architecture. These recognition architectures primarily consist of Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs). These frameworks will be discussed in detail throughout the paper. Once the model understands the image, it will extrapolate features to form the path and the behavior of the AV. This is directly fed to the hardware to change variables such as steering and acceleration [(Prakash, 2021)]. These intelligent systems induce suggestions to upgrade algorithms and pattern recognition. The reduction of concentrated workloads on one sector of the model enables a more accurate and precise decision. The individual layers and components that share its workload are able to extract patterns and adapt accordingly to enhance split-second decision-making.~~

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 models' efficiency. LiDAR sensors (Light Detection and Ranging), which are used for capturing depth in the pictures. Radars for motion detection and cameras for their enhanced ability to capture high-resolution images. These sensors benefit the object detection and semantic vision for the models [(Levinson, 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, and ViT. These frameworks neutralize the cons by increasing accuracy under computational limitations [(Selvaraju, 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 ~~were~~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 development in computer vision, new techniques like deep learning are much superior compared to their ancestors [(Khan, 2021; He, 2016)]. These mechanisms primarily consist of CNNs for semantic segmentation and object detection and ViTs for interpreting complex situations [(Khan, 2021; Vaswani, 2017)]. Software 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

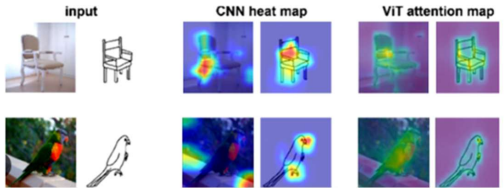
model's prediction under perturbations.

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 the traffic lights, a blur caused by

Figure 1

Figure 1. Comparison between CNN and ViT heat maps under perturbed visuals. 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. ~~Input and heat map comparison~~ [(Kang & Seo, 2024)]



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~~Some system architectures have demonstrated significant improvements and stability under these noise types. Further research into this drawback has led to the conclusion that CNNs tend to be more stable against adversarial attacks,whereas ViTs are often more stable against natural corruptions and occlusions due to their ability to leverage global image context~~. 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. e. a. Mao, 2021; Zhou, 2022)]. The recent evolution has brought the limelight to spread over computer vision; this motivates researchers to identify more robust and interpretable models for use within the AV field [(H. e. a. He, 2024; Samek, 2015)].

Computer Vision Tasks within Autonomous Systems

CVs open new doors to take on core driving assignments like lane detection, pedestrian

Formatted: Right: 0.49", Space Before: 0 pt, Line spacing: Multiple 1.73 li

identification, and sign recognition [(Janai, 2020)]. These advantages can boost models to achieve better results when it comes to abiding by the law and 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, 2017)].

~~Classical versus Deep Learning Approaches~~

~~Traditional computer vision heavily relied on manual feature classification like SIFT or ORB; these techniques are effective under structured and standardized environments. Unfortunately, when they are introduced to noise, they seem to crumble and become inaccurate [(Dalal & Triggs, 2005)]. More advanced and current models engage in these tasks with CNNs; the ability to learn hierarchical features from datasets provides a significant advantage. It has the capability to solve robust classification and detection problems [(K. e. a. He, 2016)]. Another deep learning architecture is ViT. This newly developed algorithm is able to reason with global features while maintaining performance under perturbations [(Naseer, 2021)].~~

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 is a domain that will be positively affected as transparency in an architecture becomes more abundant [(Samek, 2015)]. Figure 1. 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 colour 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, 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. e. a. He, 2024)]. ~~The lack of transparency increases every day in advanced CV models no matter the domain. Interpreting and understanding this void within the models is the most efficient method to develop~~

a truly trustworthy system. In this paper transparency techniques are considered complementary, providing context to the robustness analysis. Autonomous vehicles is a domain that will be positively affected as transparency in an architecture becomes more abundant [(Samek, 2015)]. Advancements in CV model's lucidity ensure traceability and trust within clients, additionally offering a more reliable and consistent model. Figure 1. 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 colour 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, 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. e. a. He, 2024)]. This method of gaining clarity

~~within these models contributes to the development and the usability of these advanced computational techniques.~~

Formatted: Indent: Left: 0"

Foundations of AI in Visual Perception Models

Formatted: Normal, Indent: Left: 0", Space After: 1.3 pt, Line spacing: Multiple 1.73 li

This section exposes 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 (LeCun, Bengio, & Hinton, 2015). CNNs scan the image thoroughly in patches or locally, but ViTs understand the image holistically at a global level (Dosovitskiy et al., 2020). Loss Functions identifies differences between the model's output values, it provides feedback on the precision of the system such that lower loss is higher performance (Goodfellow, Bengio, & Courville, 2016). These steps allows 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).

~~This section explores the basic principles behind Artificial Intelligence (AI) models that are involved in computer vision. The learning and development happen through layers, learning methods, and loss evaluation. These are mechanics that support CNNs and ViTs in interpreting visual data for autonomous vehicles.~~

~~Layers~~

~~Layers are like decomposers; they break down an image into different sectors and analyze them to identify objects and segment parts. Their tasks can vary from basic edge separation to element detection. CNNs do a detailed scan of the image in parts or locally. On the contrary, ViTs perceive the image by understanding the whole image at a global stage. These dynamic differences in the architecture allow a wide range of outputs that these models exhibit.~~

~~Gradient Descent~~

~~This is effectively a feedback system where models learn from their errors by comparing outputs against expected results. After each prediction, the model makes small adjustments to its internal parameters to reduce future mistakes and misclassifications. This continuous adjustment process allows the model to steadily refine its pattern recognition and improve its inference over time.~~

~~Loss Functions~~

~~This algorithm measures the variation between the model's predicted values and the expected values. This acts as a measurement tool for trained models; they provide extremely helpful~~

~~Figure 2~~

~~*Convolution Neural Network architecture*~~

~~insights into the precision rates of the system. Lower loss shows that the model is performing better or as expected. This is a major step that all models go through before deployment to verify that balance and precision are up to state-of-the-art standards.~~

~~Show of Knowledge~~

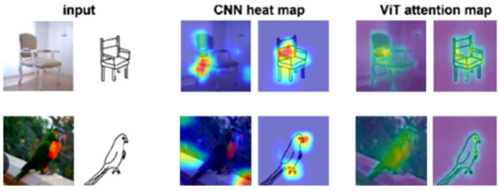
~~CNNs and ViT frameworks use these basic graphs and values to improve the models' performance during training. Due to ViTs' recent growth, they are able to handle and provide significantly more accurate answers for complex scenes than the older CNN models. These evolution metrics are standardized as benchmarks for these algorithms for comparative and comprehensible purposes.~~

Structural Comparison: CNNs and ViTs

This section 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 solve a common problem expose different perspectives and solutions, causing the dynamic trade-offs between these techniques.

Figure 1

Figure 1. Comparison between CNN and ViT heat maps under perturbed visuals. 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)]



Formatted: Normal

Figure 2
A Convolutional Neural Network (CNN) processes an image through layers of filters, pooling, and connections to recognize patterns and make predictions.

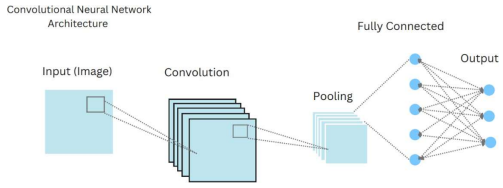
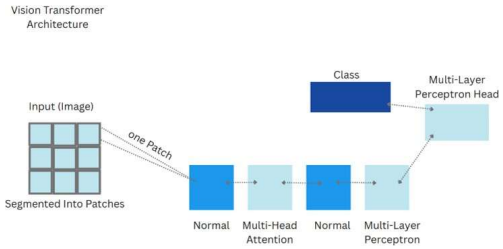


Figure 3

~~*Vision Transformer architecture*~~ *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; 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, 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 [(Girshick, 2014; Ren, 2015)]; and finally YOLO, an object detection system that uses predicting systems in bounding boxes and class probabilities directly from the full image [(Redmon, 2017)]. Figure 2 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, and they are mixed up to synthesize patterns; this is generally referred to as token mixing. An example of a ViT architecture-based model is DETR (Detection Transformers); this workflow uses encoders and decoders to help understand images [(Carion, 2020)]. Figure 3 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, 2021)].~~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; 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, 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 [(Girshick, 2014; Ren, 2015)]; and finally YOLO, an object detection system that uses predicting systems in bounding boxes and class probabilities directly from the full image [(Redmon, 2017)]. Figure 2 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, and they are mixed up to synthesize patterns; this is generally referred to as token mixing. An example of a ViT architecture-based model is DETR (Detection Transformers); this workflow uses encoders and decoders to help understand images [(Carion, 2020)]. Figure 3 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, 2021)].~~

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, 2017)]. This interpretation map allows development of a transparent model that can be trusted. Correspondingly, ViTs use self-attention maps to show 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; this is demonstrated by the consistent and accurate attention maps under the influence of noise [(Chefer, 2021; X. e. a. Mao, 2021; Zhou, 2022)]. While these interpretability differences are important, robustness metrics remain the primary lens of evaluation in this study.

~~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, 2017)]. This interpretation map allows development of a transparent model that can be trusted. Correspondingly, ViTs use self-attention maps to show 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; this is demonstrated by the consistent and accurate attention maps under the influence of noise [(Chefer, 2021; X. e. a. Mao, 2021; Zhou, 2022)]. While these interpretability differences are important, robustness metrics remain the primary lens of evaluation in this study.~~

Safety Risks in AI-Driven AVs

AI-based autonomous vehicle crash data is visualized in Figure 4 [Tesla – Williston, Florida

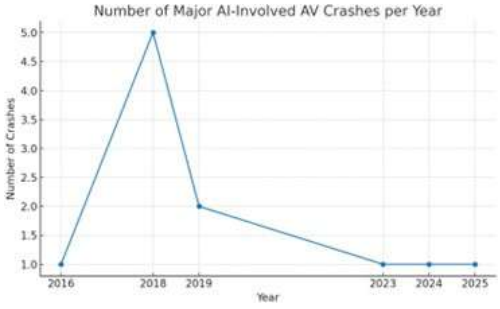
(2016) Uber ATG – Tempe, Arizona (2018) Tesla – Mountain View, California (2018) Tesla – Culver City, California (2018) Tesla – South Jordan, Utah (2018) Tesla – Laguna Beach, California (2018) Tesla – Delray Beach, Florida (2019) Tesla – Gardena, California (2019) Cruise Robotaxi – San Francisco, California (2023) Waymo – San Francisco, California (2024) Zoox – Las Vegas, Nevada (2025)]. The data was collected from public regulatory and investigation reports or example 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, presents accidents from 2010 - 2025, shows an increase between 2016 - 2019, due to the large-scale autonomous testing [(California DMV, 2022; 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. Additonally the trend in Figure 4 aligns with reported studies that show most AI-related crashes occurred during early testing phases, and a reduction 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.

Figure 4

Line graph that illustrates the number of AI-involved AV crashes per year



Additionally, the fusion of different perspectives provided by sensors (LiDAR, radar, and camera inputs) [(Levinson, 2011)] strengthened object detection reliability, though limitations still existed. The first fatality recorded in a fully autonomous system [(NTSB, 2019)] occurred when the Uber ATG algorithm misclassified a pedestrian multiple times within six seconds. 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. It developed fear and caution within the AV space. In the next sections of this paper, we will cover all factors and solutions to prevent fatalities along with lowering the graph's accident frequency.~~AI-based autonomous vehicle crash data is visualized in Figure 4 [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)]. The data, representing accidents from 2010 to 2025, shows a spike between 2016 and 2019, coinciding with the large-scale autonomous public testing [(California DMV, 2022; NTSB, 2020)]. The gradual decrease after regulations were synthesized resulted in a dynamic decline, leading to a stable and relatively low frequency of accidents. This is achieved by intensive testing in evaluation and training stages; additionally, computational simulations and~~

~~metrics like mean average precision were used before public examinations. Risk factors like adverse weather and low-light contrast demonstrated instability in the model.~~

~~Figure 4~~

~~*Line graph that illustrates the number of AI-involved AV crashes per year*~~

~~Additionally, the fusion of different perspectives provided by sensors (LiDAR, radar, and camera inputs) [(Levinson, 2011)]. The first fatality recorded in a fully autonomous system [(NTSB, 2019)], an Uber ATG algorithm misclassified pedestrians' multiple times within 6 seconds. Resulting in an unstable path prediction, this caused the emergency brake system to disable due to a false-positive decision. A pedestrian was killed in Tempe, Arizona, due to the failure of the model classification techniques. It developed fear within the AV space. In the next sections in this paper, we will cover all factors and solutions to prevent fatalities along with lowering the graphs' accident frequency.~~

Types of Noise

During training noise is intentionally injected in 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 the road block [(Dosovitskiy et al., 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; Chen et al., 2017)]. Preprocessing will remove environmental distortions ensuring that visual inputs are consistent, this is crucial as both ViTs and CNNs depend on clean images.

Formatted: Font: Not Bold

Formatted: Font: Not Bold

Formatted: Font: Not Bold

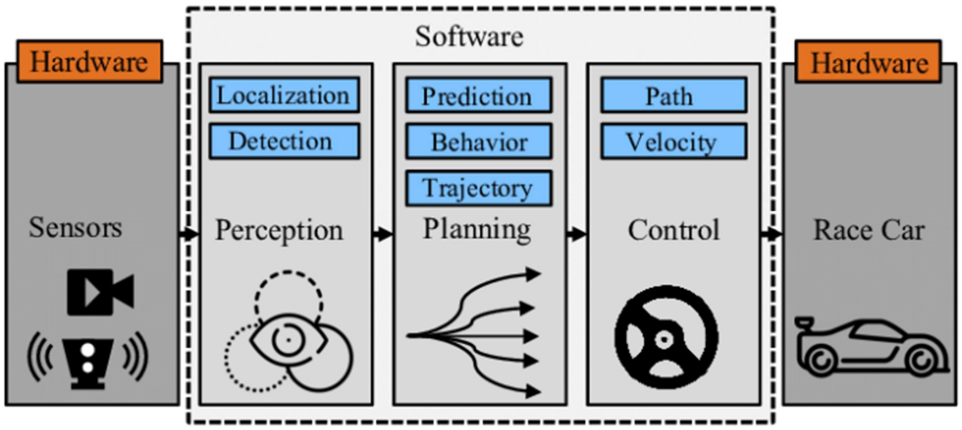
Formatted: Font: Not Bold

Formatted: Font: Not Bold

Formatted: Not Expanded by / Condensed by

Formatted: Normal

Figure 5. A simplified perception pipeline representing how and input image from the LiDAR, radar or camera travels through a model (either CNN or ViT), with a layers that allow for noise mitigation. Additionally, it describes the process of decoding an image in three sections; perception, planning, and control. (*Autonomous Vehicles on the Edge: A Survey on Autonomous Vehicle Racing – Scientific Figure on ResearchGate.* (n.d.). Retrieved October 9, 2025, from https://www.researchgate.net/figure/Autonomous-driving-pipeline-including-both-hardware-and-software-that-provides-the_fig1_361243142)



Motion and Gaussian Blur

Blurs have the ability to hide or distort crucial information; the lack of clarity in vision can allow models to overlook key aspects that influence the interpretation. Swift movements can cause motion blur; Figure 5 provides a visual example from AIEase (n.d.). Free AI motion blur effect online. AIEase (an online blurring tool). These blurs decreases the interpretable information available for the model [(Nah, 2017)]. Identically Gaussian blurs are synthetic simulations of elements that can distract a model; these are often used in benchmark datasets to analyze

Formatted: Font: Not Bold
Formatted: Font: Not Bold
Formatted: Font: Not Bold
Formatted: Font: Not Bold
Formatted: Not Expanded by / Condensed by
Formatted: Normal

Figure 5

Shows how motion and Gaussian blur distort a picture. Fast movement or synthetic blur hides important details and confuses the model when detecting objects ~~*Motion and Gaussian Blur*~~

situation-based noises [(Hendrycks & Dietterich, 2019)].

Fog and Haze

Fog adds scattered lighting, creating luminous areas that can wash out color and contrast in an image. The Figure 6 provides an idea of how much detail this perturbation can remove [(Henson, 2022)]. Detecting entities can become a big trouble under this type of noise; dehazing is a necessary concept to make data ~~understandable~~ predictable [(Li, 2017)]. AOD-Net techniques provide a great network to prevent this type of error, allowing for quick and easy haze removal [(Li, 2017)].

Rain & Atmospheric Noise

As mentioned in the previous section, a very common and natural noise is rain. These create long, high-frequency streaks and varied lighting on artifacts. Figure 7 visualizes the noise [(LR-PS Tutorials, n.d.)]. Detailed restoration networks can mitigate these disturbances; clean~~s~~ing techniques can preserve content while removing rain or any weather-based anomalies [(Fu, 2017)]. Rain-specific training allows models to ignore strokes in the image during the classification phase [(Zhang, 2019)].

Figure 6

Shows how fog removes color and contrast, washing out the picture. The model loses small details and can miss objects on the road.~~Noise due to fog~~

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 8 provides a description and results when occlusions are present [(Ryu & Chung, 2021)]. This can prevent models from getting a global understanding of the context presented; it also contributes to object misclassification [(Hendrycks & Dietterich, 2019)]. 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)].

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 9 shows a low-contrast environment; the loss of clarity and detail is portrayed by the change in lighting [(Galer, n.d.)]. Texture-sensitive models are mainly affected by this noise [(Liu, 2017)]. To combat this, the addition of preprocessing methods like exposure corrections and dynamic range compensation is deemed necessary [(Chen, 2018)].

Figure 7

Rain creates streaks and light changes that block vision. The blurred parts make it harder for the model to see objects correctly. ~~Blur caused by Rain~~

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 limitation heavily influences the model's interpretability. This section addresses solutions to solve these imperfections, methods like pixel-level cleaning, deblurring and visibility enhancements, weather-specific processing, and robust training mitigate these perturbations. These approaches collectively improve clarity and preserve detailed information for answering prompts; additionally, they develop resilience against perturbations in images.

Pixel-Level Cleaning

Images tend to have different variations of light contrast; this can cause dynamic changes in interpretation as it makes it hard for models to split into layers and understand the global idea. The usage of histogram equalizations or CLAHE (Contrast Limited Adaptive Histogram Equalization) mitigates these risks. They spread out pixel intensity values to ensure darker regions become brighter and lighter regions become dimmer. This improves the global contrast of the picture, giving the model a genuine perspective of the image [(García, 2017; 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 ~~when the model is able to compare sizes~~ compares sizes of objects to synthesize patterns, successfully adding to the

Figure 8

Example of how blocking objects hide what's behind them. It shows how occlusion can confuse the model by cutting off parts of the image. ~~*Examples depicting occlusion*~~

(a) Occlusion by Preceding Vehicles (b) Occlusion by Shadow and Truncation

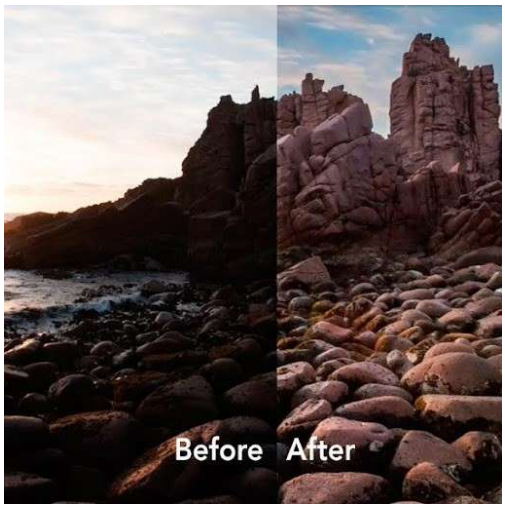
model's predictions and output [(Howard, 2017)].

Deblurring and Visibility Enhancement Methods

Blurs and visibilities have affected ~~our~~ human sights, causing inaccurate judgments and a lack of understanding of ~~our~~ the surroundings. Figure 10 "What Is Imerge Pro?" (Manula, FXhome, 2021) 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 our eyes 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, 2017)]. All-in-One Dehazing (AOD) is a technique used to cleanse a picture from perturbations in a singular and contained space. Figure 11 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; it aids in the restoration of a sharp image [(Li, 2017)].

Figure 9

Shows how lighting changes between bright and dark areas confuse the model. Loss of texture and contrast reduces what it can detect. ~~Difference in illumination~~



Weather-Specific and Rain Removal Pre-processing

Raindrops create streaks in images, acting as disturbances ~~distractions~~ 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 its significance and eradicate the effect~~This negatively impacts the model's stability; therefore, techniques like Detail-Recovery Network (DNN) eradicate it~~. DNNs use high-frequency isolations, where rain strokes usually appear, to remove these stripes. High-pass filters are applied to input images to carry out this procedure; continuous training on these specific perturbation models develops resilience toward it. Loss functions are integrated to balance rain removal and detail preservation; this can help preserve fine edges in a figure [(Fu, 2017)]. Other weather-based anomalies in an image are treated using weather-augmented training. It 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. [(Zhang, 2019)]

Figure 10

Shows a clear picture beside a blurred one. You can see how much detail is lost when blur appears, lowering model accuracy. ~~Motion and Gaussian Blur~~

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 robust 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, 2019). Although adding anomalies strengthens the model's interpretability, it fails to provide statistical evidence to fully address this concept's validity. A more developed yet naive approach involves weaving corrupted images and clean ones 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, 2019)].

Figure 11

Shows how the AOD-Net method clears haze from an image. After cleaning, the picture becomes sharper and easier for models to understand. ~~AoD Motion and Gaussian Blur Removal~~

(a) Haze (b) AOD-Net

Model Strengths Across Perturbation Scenarios

The diversity in vision model architecture ~~architects~~ allows 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, 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; also, they were able to adapt to partial occlusions. The strengths describe the following: s~~S~~tability ensures reliable and balanced performance,~~,~~ r~~R~~etention provides durable and strong memory.

Robustness reflects the adaptability of the algorithm. Coverage delivers the wide scope of the analyzed system,~~,~~ d~~D~~etection enables consistent recognition,~~,~~ r~~R~~esilience shows the flexible and persistent nature of the models,~~,~~ c~~C~~onfidence refers to trustability, and adaptability ensures versatile and evolving capacity.

Methods

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 the broader and more reliable

Formatted: Font: 12 pt

view, since the values are taken from peer-reviewed empirical papers rather than single controlled runs.

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:

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)].

$$\mu = \frac{1}{N}\sum_{i=1}^{N}\frac{mAp_i - mAp_{Clean}}{mAp_{Clean}}$$

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 et al., 2025)].

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

min-rPC (worst-case) portrays 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 et al., 2021)].

$$\text{min-}-rPC = \min\left(\frac{mAP_{Perturbed},i}{mAp_{clean}} - \mu\right)$$

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

Each model's clean and perturbed accuracy values were obtained from verified sources. The computation

Collecting clean and corrupted mAP(mean average precision) results from the benchmark

Formatted: Indent: Left: 0.5", First line: 0"

Formatted: Indent: Left: 0.5", First line: 0"

Formatted: Indent: Left: 0.5", First line: 0"

Formatted: Font: Cambria Math

Formatted: Font: Cambria Math

datasets.

Normalizing results across the architectures for consistent comparison.

Computing μ, σ, and min-rPC for each natural perturbation type.

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 the models used in this study follows 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 [(He et al., 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 et al., 2020; Mao et al., 2023; Liu et al., 2021)].

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, 2019; Lin et al., 2014)], in addition to robustness was

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)].

The studies used in this secondary analysis were filtered through a structured literature screening process. Searches were done using Google Scholar, IEEE Xplore, arXiv, and SpringerLink using 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. Papers that used standardized benchmarks such as COCO or ImageNet-C were prioratized and clearly specified noise types and evaluation metrics. This ensured methodological consistency across compared results and minimized dataset or reporting bias.

Formatted: Normal, Indent: First line: 0.5", Right: 0.49", Space Before: 0 pt

Observation & Analysis

This section aims to provide a secondary analysis on CNNs' and ViT-based models' 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 diferent 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; Mao et al., 2023)]. It is the most used dataset for evaluating object segmentation and detection, additionally it forms the basis for all mAP (mean average

precision) 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, 2019; Mao et al., 2023)] covering both environmental and synthetic distortions typically encountered in road scenes.

Formatted: Right: 0.77"

Table 1

Model strengths across different perturbation scenarios based on the literature, Table 1 is composed of data derived from secondary analysis on existing literature as mentioned below. This data consists of different types of noise and the corresponding models that exhibit the best accuracy for each type of noise. This allows for a comparative analysis of the architectural resistance to distinct types of perturbations.

Perturbation Type	**Scenario**	**Best Model**	**Strength**	**Citations**
Environmental	Moderate rain	CNN	Stability	Fu et al., 2017; Zhang et al., 2019
Environmental	Fog	CNN	Retention	Li et al., 2017; Henson, 2022
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, 2019; Mao et al., 2021
Natural	Shadows	ViT	Resilience	Chefer et al., 202
Adversarial	Pixel noise	ViT	Confidence	Mao et al., 2021; Paul & Chen, 2021
Occlusion	Partial blockage	Hybrid CNN+ViT	Adaptability	Ryu & Chung, 2021; Ghiasi et al., 2018

~~Table 1~~

~~*Model strengths across different perturbation scenarios based on the literature*~~

~~**Perturbation Type**~~	~~**Scenario**~~	~~**Best Model**~~	~~**Strength**~~
~~Environmental~~	~~Moderate rain~~	~~CNN~~	~~Stability~~
~~Environmental~~	~~Fog~~	~~CNN~~	~~Retention~~
~~Environmental~~	~~Low-light glare~~	~~ViT~~	~~Robustness~~

~~Natural~~	~~Snow~~	~~Hybrid CNN+ViT~~	~~Coverage~~
~~Natural~~	~~Dust~~	~~Hybrid CNN+ViT~~	~~Detection~~
~~Natural~~	~~Shadows~~	~~ViT~~	~~Resilience~~
~~Adversarial~~	~~Pixel noise~~	~~ViT~~	~~Confidence~~
~~Occlusion~~	~~Partial blockage~~	~~Hybrid CNN+ViT~~	~~Adaptability~~

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 below in Table 2. The models selected for comparison are: Faster R-CNN, Mask R-CNN, EfficientDet-D4, DETR, Deformable DETR, DINO, and Swin-L. They were chosen based on their leading benchmark architectures for object detection [(Ren et al., 2015; Carion et al., 2020; Zhou et al., 2022)]. Each model captures a different design: region-based proposals (CNNs) against transformer-based global attention (ViTs). They enable a balanced cross-architecture analysis. ~~This subsection supports these claims by providing relevant statistics. To recap, CNN's main defect is its vulnerability to dynamic perturbation; they often fail under low contrast or high blur conditions due to reliance on local spatial features [(X. e. a. Mao, 2021)]. ViTs, on the other hand, perform better under disturbance due to global and self-attention mechanisms [(Zhou, 2022)]. The COCO benchmark dataset, includes a wide range of perturbations for CV tasks where Gaussian Blur could cause up to 45% mAP reduction (Mean Average Precision scores; they are used to compare and rate models, and their values range from 0 to 1 [(X. e. a. Mao, 2021)]) as compared to 25% for ViT architecture systems.~~

Formatted: Right: 0.51", Space Before: 0 pt

Commented [AA1]: Check with uncle if I need to change this

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, allowing both detection accuracy and completeness [(Hendrycks & Dietterich, 2019; 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,

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, 2021)]
Mask R-CNN	CNN	Natural	63.1	49.4	13.7	[(Shankar, 2021)]
EfficientDet-D4	CNN	Natural	49.4	~35	14.4	[(Zhou, 2022)]
DETR	ViT	Natural	42.0	~32	10.0	[(C. e. a. Mao, 2023)]
Deformable DETR	ViT	Natural	45.4	~34	11.4	[(C. e. a. Mao, 2023)]
DINO (Swin-L)	ViT	Natural	51.3	~38	13.3	[(Zhou, 2022)]
Swin-L Detector	ViT	Natural	58.7	~44	14.7	[(Zhou, 2022)]

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, 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 they 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 [(Levinson et al., 2011; Bojarski et al., 2016)].

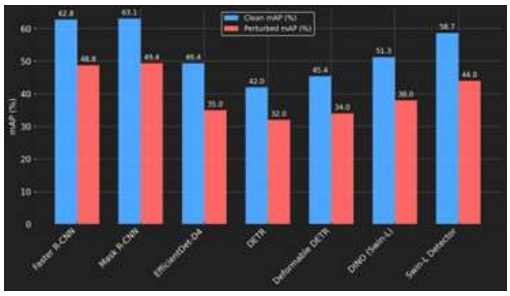
Key Observations & Analysis

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; the

Figure 12

Graph comparing model accuracy on clean images versus noisy ones. Every model drops in performance when exposed to natural distortions. ~~*Clean vs perturbed mAP under natural perturbations*~~



drops vary 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.

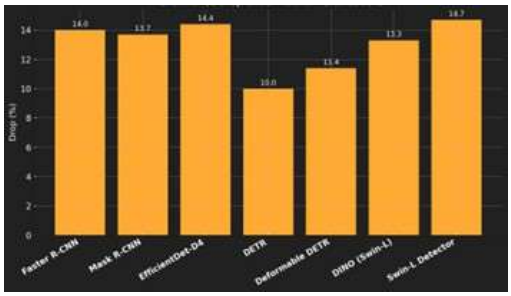
CNN Models: Robustness in Certain Scenarios

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; Table 2 supports this argument. EfficientDet-D4, regardless of 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 show 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.

Figure 13

Shows how much each model's accuracy falls because of noise. ViTs lose less accuracy than CNNs, proving stronger stability. ~~*Performance drop under natural perturbations*~~



Higher Accuracy Does Not Guarantee Robustness

Although Swin-L detectors have a clean mAP of 58.7%, it falls to ~44% under defects, this proves that a higher baseline precision does not refer to better perturbation resistivity. Similarly, another model with a relatable issue is EfficientDet-D4, where it holds a modest clean mAP yet suffers a comparable drop to the state-of-the-art CNNs and ViTs. This underlines 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 Tabel 2 corresponded to natural perturbations like motion blur and fog; this ensures 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.

Advanced Robustness Analysis

Traditional graphs (Figure 13,12) illustrate clean vs perturbed mAP; this provides an understanding of the “average” performance. While these insights are helpful, these face-value

Table 3

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. ~~*Reliability distributions for CNNs versus ViTs.*~~

Formatted: Indent: Left: 0", Right: 0.49", Space Before: 0 pt, Line spacing: Multiple 1.73 li

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

perspectives hide critical weak points in the algorithm. A model can appear strong overall, yet crumble under corruptions in an image. It is an important constraint for AV tasks because the model deployment is delayed by rare but harmful failures [(Michaelis, 2019), (Wang et al., 2024)].

Multi-Metric Robustness Framework

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.

~~Metrics~~

~~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)].~~

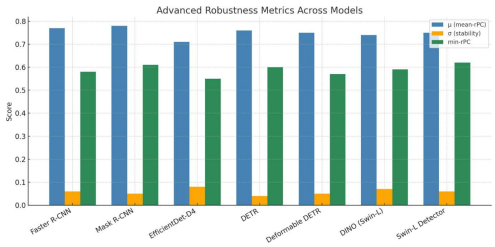
$$\mu = \frac{1}{n}\sum_{i=1}^{N} mAP_i - mAP_{clean}$$

mAP_{clean}

(1)

Figure 14

Bar graph comparing three robustness metrics for all models. CNNs score higher on average strength, while ViTs show more stability and better worst-case results.~~Bar graph illustrating the advanced robustness metrics (μ, σ, min-rPC) for the selected models.~~



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 et al., 2025)].~~

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

~~min-rPC (worst-case) portrays 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 et al., 2021)].~~

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

~~Table 3 demonstrates reliability distributions for CNNs versus ViTs, highlighting that CNNs vary more widely between disturbances, while ViTs focus on stability, compromising on averages.~~

Formatted: Space Before: 0 pt

The bar graph in Figure 14 displays the three advanced parameters as mentioned above (μ, σ, min-rPC) for all 8 of the models. Figure 14 indicates CNNs' and ViTs' average performance, accuracy, and safety under noise. ViTs indicate trends, with lower σ metrics (0.04 for DETR); this shows greater accuracy. Although this is good, it simply shows a low min-rPC (0.60). Models like the Swin-L Detector have optimal worst-case robustness (0.62) and reasonable average robustness (0.75). CNNs, on the other hand, show greater μ (mean-rPC) on

average, fluctuating from 0.71 to 0.78, but algorithms like EfficientDet-D4 suffer from high volatility ($\sigma = 0.08$). DINO Swin-L performs consistently but does not beat the Swin-L Detector. The graph reveals that CNNs lead in average resilience, but ViTs provide safer lower limits. This is a crucial component that is vital for AV security.

Mask R-CNN shows a higher mean robustness ($\mu = 0.78$) and accuracy σ (0.05) when compared to Faster R-CNN and EfficientDet-D4. On a contrast basis, it outperforms Faster R-CNN in worst-case robustness by 0.03, this proves its supremacy in the design, balancing the accuracy and dependability, making 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.

Discussion and Conclusion~~Discussion~~

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, features in detailed and structured environments [(LeCun, 1998; He et al., 2016)]. 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 [(Mao et al., 2021)]. On the other hand, ViTs are built to understand images globally through self-attention

mechanisms [(Dosovitskiy et al., 2021; Zhou et al., 2022)]. 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 et al., 2021; Samek et al., 2015)]. CNNs build more interpretable Grad-CAM heatmaps that make it clear where the model is focusing, where as 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 recourses, 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. 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 and hide the internal inconsistencies. Advanced metrics such as mean-rPC, stability σ, 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 percent mAP drop for DETR and 13 to 14 percent 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. ~~The comparative study of ViTs and CNNs provides insights into the frameworks' trade-offs that could potentially affect the algorithms' interpretability, robustness, and practicality in autonomous driving tasks. Both the architectures have significant benefits in different sectors of the domain. CNNs have greater potential to excel in local texture recognition; this allows them to achieve strong performance values under heavy detail-oriented and structured environments.~~

~~Unfortunately, they struggle to maintain this under distortions that span the whole visual input; disturbances like fog and low-light contrast are some examples. On the contrary, ViTs surpass CNNs in global understanding and self-attention. They provide reliable outputs when they are~~

under the influence of a large range of distortions. Yet the computational cost and latency factors drag the deployment of these architecture-based models. Additionally, interpretability creates another layer that separates these systems; CNNs are able to provide more intuitive Grad-CAM maps. This enables transparency and reveals influential regions in an image. However, ViTs are not able to match its opaque nature; the attention scores and segment-wise analysis make ViTs harder to understand, thereby reducing visibility in their decisions. The fundamental concept that allows these networks to make predictions is the dataset and the algorithm; they are the most significant components in a system. Therefore, benchmark datasets that exaggerate a model's stability must not be taken for analysis based on the face value. One main reason is that these databases cannot mimic the unpredictable real-world conditions. While the algorithms have enhanced significantly over the past years, hardware effectiveness becomes apparent as an understudied factor. Although ViTs have the capacity to outperform other architectures in robustness, CNNs still dominate in cost-effectiveness in resource-driven AV systems.

Future Steps

The next stage of development in the model must focus on integrating and adapting across algorithms rather than constraining the scope to isolated architecture comparison. There must be a dynamic switch from developing secluded systems to hybrid perception models that have the ability to adapt to diverse real-world noises. Coordinating various sensors like LiDAR and radar to simultaneously work with vision algorithms can enhance outputs; this eradicates the dependence on cameras. Although they have the capacity to illustrate the full picture, they are the most fragile when it comes to resilience in visual perturbation capture. The dynamic increase in the development of vision algorithms is influencing the domain significantly, but there are no standardized metrics to measure robustness accurately. To enable comparison, conventional measurements must be used to bring out the true potential of models. Current systems are trained on exaggerated perturbations to ensure maximum robustness. Yet this training method fails to

address the unpredictable and realistic conditions that humans come across every day. Establishing a real-world scenario-based database exposes the real-time performance and will allow models to find applicable patterns.

Conclusion

While CNNs and ViTs thrive in the field of autonomous vehicles, independently, they cannot provide absolutely reliable and consistent outputs. Even though CNNs offer full disclosure for decisions and accuracy, but they often suffer under various distortions in real-world scenarios. Faster R-CNN and Mask R-CNN lose over 13 to 14% of their accuracy when they face noise, as recorded in the paper, proving that they are unstable under noise. On the contrary, ViTs provide stability under perturbations; however, they lack interpretability and energy efficiency in real-world AV integrations. DETR shows the lowest drop in mAP score (~10%); additionally, Swin-L has the highest clean accuracy (58.7%), yet these advantages are dragged down due to its complex structure. Robustness should not be solely dependent on the model; the responsibility must be shared across the entire pipeline. Every component, like preprocessing and model framework, has an influence on the performance. Further analysis reveals that a hybrid pipeline that integrates both architectures can preserve performance under granular noise while maintaining accessibility. Therefore, they dominate single-architecture approaches in AV tasks. Ultimately models like EfficientDet-D4 only record a worst-case rPC of 0.55; this is the lowest score recorded by this paper. This rises as a pressing issue due to the rare but high-risk failure factor experienced by underperforming algorithms, making it a challenge to adapt models in real-world cases. The most promising path ahead is to adapt to hybrid architectures that include enhanced cross-sensor collaboration and methods to mitigate noise and evaluation benchmark datasets. This enables trustworthy, explainable, and resilient AV systems to attain a new perspective of the available architectures. Uniting these factors and architectures synthesizes a comprehensive and safe deployment of autonomous algorithms to strengthen driving tasks.

References

Badue, C. e. a. (2021). Self-driving cars: A survey. *Expert Systems with Applications*, *163*, 113791.

Bhojanapalli, S. e. a. (2021). Understanding robustness of transformers for image classification. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021*.

Bojarski, M. e. a. (2016). End-to-end learning for self-driving cars. *arXiv:1604.07316*.

Caesar, H. e. a. (2020). Nuscenes: A multimodal dataset for autonomous driving. *Computer Vision and Pattern Recognition (CVPR)*.

California DMV. (2022). Autonomous vehicle disengagement reports 2022. *Report*.

Carion, N. e. a. (2020). End-to-end object detection with transformers. *European Conference on Computer Vision (ECCV)*.

Chefer, H. e. a. (2020). Transformer interpretability beyond attention visualization. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Chefer, H. e. a. (2021). Transformer interpretability beyond attention visualization. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021)*.

Chen, Z. e. a. (2018). An attention-based deep learning framework for low-light image enhancement. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Cordts, M. e. a. (2016). The cityscapes dataset for semantic urban scene understanding. *Computer Vision and Pattern Recognition (CVPR)*.

Dai, J. e. a. (2017). Deformable convolutional networks. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*.

Dong, Z., Xu, X., He, S., Wu, Z., Xie, J., & Chen, T. (2025). Tire slip angle estimation based lateral stability control strategy for trajectory tracking scenarios of distributed drive

autonomous electric vehicles. *Control Engineering Practice*. https://doi.org/10.1016/j.conengprac.2025.106343

~~Dosovitskiy, A. e. a. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations (ICLR)*.~~

Fu, X. e. a. (2017). Removing rain streaks from a single image via deep detail network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Galer, M. (n.d.). Develop: Editing images with extreme contrast [free lightroom tutorial] [n.d.].

~~García, N. e. a. (2017).~~

Ghiasi, G. e. a. (2018). Dropblock: A regularization method for convolutional networks. *Advances in Neural Information Processing Systems (NeurIPS)*.

Girshick, R. e. a. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

He, H. e. a. (2024). Cf-cam: Cluster filter class activation mapping for reliable gradient-based interpretability. *Pattern Recognition*.

He, K. e. a. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hendrycks, D., & Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. *International Conference on Learning Representations (ICLR)*.

Henson, T. (2022, January). Before & after: Foggy morning on casco bay [[Photograph]. Todd Henson Photography].

Howard, A. e. a. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv:1704.04861*.

Janai, J. e. a. (2020). Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends in Computer Graphics and Vision*.

Kalra, N., & Paddock, S. (2016). Driving toward a wiser tomorrow. *RAND Corporation*.

Kang, S., & Seo, K. (2024). Sketch classification using vit. *JEET*, *19*, 4587–4593.

Khan, S. e. a. (2021). Transformers in vision: A survey. *ACM Computing Surveys*.

Lai-Dang, T. (2024). Interpretable medical imagery diagnosis with self-attentive transformers: A review of explainable ai for health care. *BioMedInformatics, 2024*.

Lambertenghi, G. e. a. (2025). Benchmarking image perturbations for testing automated driving assistance systems. *Proceedings of the IEEE International Conference on Software Testing, Verification and Validation (ICST 2025)*.

~~LeCun, Y. e. a. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278–2324.~~

Levinson, J. e. a. (2011). Towards fully autonomous driving: Systems and algorithms. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*.

Li, B. e. a. (2017). Aod-net: All-in-one dehazing network. *International Conference on Computer Vision (ICCV)*.

Liu, Y. e. a. (2017). Learning a deep single image brightening network with attention-based feature fusion. *IEEE International Conference on Image Processing (ICIP)*.

LR-PS Tutorials. (n.d.). Add a rain effect to the photo [n.d.].

Mao, C. e. a. (2023). Coco-o: A benchmark for object detectors under natural distribution shifts. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Mao, X. e. a. (2021). Adversarial attacks are reversible with natural supervision. *In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (pp. 641–651)*.

Michaelis, C. e. a. (2019). Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*.

Nah, S. e. a. (2017). Deep multi-scale convolutional neural network for image deblurring. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

~~Naseer, M. e. a. (2021). Intriguing properties of vision transformers. *Advances in Neural Information Processing Systems (NeurIPS)*.~~

NTSB. (2019). Collision between vehicle controlled by developmental automated driving system and pedestrian, tempe, arizona, march 18, 2018. *Report*.

NTSB. (2020). Collision between vehicle controlled by developmental automated driving system and pedestrian, tempe, arizona, march 18, 2018. *Report*.

Paul, S., & Chen, Y. (2021). Vision transformers are robust learners. *In Proceedings of the AAAI Conference on Artificial Intelligence*.

Prakash, A. e. a. (2021). Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Redmon, J. e. a. (2017). Yolo9000: Better, faster, stronger. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ren, S. e. a. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems (NIPS)*.

Ryu, S.-E., & Chung, K.-Y. (2021). Appl. sci. 11(15), 7093. *Journal*.

Samek, W. e. a. (2015). Evaluating the visualization of what a deep neural network has learned. *International Conference on Learning Representations (ICLR) Workshop*.

Selvaraju, R. e. a. (2017). Grad-cam: Why did you say that? *International Conference on Computer Vision (ICCV)*.

Shankar, V. e. a. (2021). Do image classifiers generalize across time? *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 9661–9669.

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, *6*(1), 1–48.

Subbaswamy, A., Adams, R., & Saria, S. (2021). Evaluating model robustness and stability to dataset shift. *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*. https://proceedings.mlr.press/v130/subbaswamy21a.html

Szegedy, C. e. a. (2015). Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

~~Thrun, S. (2010). Toward robotic cars. *Science*, *327*(5969), 1215–1215.~~

Vaswani, A. e. a. (2017). Attention is all you need. *Advances in Neural Information Processing Systems (NIPS)*.

Wang, S., Yu, Z., Jiang, X., Lan, S., Shi, M., Chang, N., Kautz, J., Li, Y., & Alvarez, J. M. (2024). Omnidrive: A holistic llm-agent framework for autonomous driving with 3d perception, reasoning and planning. *arXiv preprint arXiv:2405.01533*.

Yi, C., Yang, S., Li, H., Tan, Y.-P., & Kot, A. C. (2021). Benchmarking the robustness of spatial-temporal models against corruptions. *Advances in Neural Information Processing Systems (NeurIPS)*.

Yu, T. e. a. (2020). Bdd100k: A large-scale diverse driving video dataset. *Computer Vision and Pattern Recognition (CVPR)*.

Zhang, H. e. a. (2019). Image de-raining via a conditional generative adversarial network. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhou, W. e. a. (2022). Understanding the robustness in vision transformers. *arXiv:2201.03714*.

Zuiderveld, K. (1994). Contrast limited adaptive histogram equalization. *Graphics Gems IV*.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). *An Image is Worth 16×16 Words: Transformers for Image Recognition at Scale*. **arXiv preprint arXiv:2010.11929**.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). *Deep Learning*. **Nature**, 521(7553), 436–444.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., ... & Guo, B. (2021). *Swin Transformer: Hierarchical Vision Transformer using Shifted Windows*. **Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)**.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., & Jégou, H. (2021). *Training Data-Efficient Image Transformers & Distillation through Attention*. **Proceedings of the International Conference on Machine Learning (ICML)**.

Zhong, Z., Zheng, L., Kang, G., Li, S., & Yang, Y. (2017). **Random erasing data augmentation.** *arXiv preprint arXiv:1708.04896.* https://arxiv.org/abs/1708.04896

Formatted: Font: 12 pt

Formatted: Space After: 1.25 pt, Line spacing: Multiple 1.73 li

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). **ImageNet classification with deep convolutional neural networks.** *Advances in Neural Information Processing Systems (NeurIPS)*, *25*, 1097–1105. https://doi.org/10.1145/3065386

Taş, Ö., Kuhnt, F., Zöllner, J. M., & Stiller, C. (2016). **Functional system architectures towards fully automated driving.** *IEEE Intelligent Vehicles Symposium (IV)*, 304–309. https://doi.org/10.1109/IVS.2016.7535419

Formatted: Space After: 1.25 pt, Line spacing: Multiple 1.73 li

Formatted: Font: 12 pt

Cover Letter to Reviewer 1

Dear Reviewer,

I would like to thank you very much for your time in reading and reviewing my paper "Eyes on the Road: Elucidating ViTs and CNNs Under Real-World Noise."
I'm grateful for you for the constructive feedback you provided. Your feedback made me improve the paper to make it clearer, more organized, and more readable.
Below, I've mentioned what I changed in each section according to your recommendations.

Comment 1: “It is not fully clear where the results in a few of the figures come from. It appears that the authors have done some analysis on available data but this should be reported clearly and rigorously. Some more explanation on this (e.g. including an analysis methodology section) would be welcome. Also, the figures are good and rich but they could be integrated more with the main manuscript and the figure captions could be more detailed for clarity. Perhaps adding more structure in the paper to differentiate the content that is an outcome of literature review and the one that results from new analyses would enhance readability.”

Reply: (done and implemented) I have added clear captions for every figure to ensure clarity. Additionally in the abstract I have clearly stated that this paper is a secondary analysis of the empirical data present in the previous papers. I have also re-organized and sectioned the content into different level for more depth and flow.

Comment 2: “Some more insights on why CNNs are more interpretable than ViTs and how their combination would improve performance and robustness would be very informative here. Has this been implemented before or are there examples that corroborate this suggestion? What features of the algorithms underlie the differences that the author mentions and how can these be improved by the hybrid model? Ralting this interpretations back to the metrics that they used would make the point stronger and more supported.”

Reply: (incorporated most of the changes) The comparison between CNNs and ViTs section has been updated to include more detailed information on how each of these architects work and why some are better than the other in some scenarios. I have also discussed that hybrid models are significantly better and I have also reasoned it out in my discussion section.

Comment 3: “A few sentences tend to be complicated with some level of repetition. Writing could be simplified and become tighter also in relevance to the improvements in structure and figure integration I mentioned above. This would enhance readability.“

Reply:(done and implemented) longer sentences have been shortened and simplified for better flow, also diagrams have been reorganized.

Warm regards,
[Author name redacted]

Cover Letter to Reviewer 2

Dear Reviewer,

Thank you so much for your detailed and thoughtful review of my paper "Eyes on the Road: Elucidating ViTs and CNNs Under Real-World Noise. "Your feedback helped me understand what needed more explanation and how to make the paper more balanced and easier to read. Below, I've explained the changes I made based on your suggestions.

Comment 1: "**Abstract:**

- Minor comment but if you use acronyms in your abstract (like mAP) you should define them, even if you do again later in your paper
- From the abstract, it is a little unclear whether you are writing a review paper or doing an analysis. In fact, I think that this is a little unclear in your entire paper. I'd make this crystal clear in your abstract "

Reply: (done and implemented) mAP has been defined and papers goal is achived by a secondary analysis this has been stated in the abstract.

Comment 2:" **Introduction:**

- I would add a specific "Introduction" header to make the transition from abstract clear
- You spend the beginning of the introduction talking about how autonomous vehicles are impacted by the data that they receive, and then transition into discussing the algorithms you will be reviewing. However, you never explicitly link that these algorithms are making the decisions for the car. I think it would be helpful to back up a little, and spend time talking about the flow of information in this systm– first, the car records video information, then the video data is processed by algorithms (like CNNs), and based on the results of those algorithms, the car changes its behavior.
- You focus on cars that utilize image data – this is not exclusively used by all autonomous vehicles. In fact, many new cars (like you mention later) use lidar exactly as a response to the challenges of image disturbance. I think you should add a caveat earlier that improving image processing is one solution to improving car performance. This is okay, it's good scientific practice to be as specific about your conclusions as possible
- What is misclassification in this context? Even if it is obvious to you, it might not be for your reader (you should imagine young adults who aren't necessarily familiar with self-driving cars), so don't be afraid to be specific

- *"Understanding the way of predicting the accuracy is fundamental for humans to trust artificial intelligence systems in safety-critical sectors."*- this sentence is unclear to me. Maybe review for a typo
- Since your key gap is about interpretability, I would explain a little more how you think this information would be used. On one hand, you could argue that if a person is never reviewing the model outputs, it doesn't matter if the model is interpretable. Meanwhile, I agree that trust is crucial. Maybe to bolster your argument, give specific examples of how interpretability is important in building self-driving cars
- Explain more what the COCO benchmark set entails
- In general, the introduction lays out the research question and goal very clearly "

Reply: (done and implemented) Introduction was added, misclassification has been defined, and other feedback has been incorporated.

Comment 3: "Computer Vision and Autonomous Vehicles:

- Overall, I think this section is well reasoned and clear
- At times, I think your paper is a little lengthy. For example, the paragraph starting with *"This segment studies the use of Computer Vision…."* could be shortened to a one or two sentences
- I know you already have a lot of figures, but this section could be a good place for an overview schematic or graphical abstract showing the flow of information through the system. It's still a little unclear to the reader at where in the process of self-driving the algorithms that you are analyzing fit in
- The section *"How Modules Collaborate"* could be vastly reduced (or basically turned into the schematic I described earlier). In general, I don't think you need to tell me what you are discussing later in the paper, I think for holding the reader's attention, you should just get to the point succinctly
- In the *hardware* section, I don't think all cars use LiDAR sensors. Some use just cameras, like Teslas, famously. I would maybe be a more clear if you are focusing on specific types of cars
- This is the first time you use YOLO, please define or remove
- In the *Software* section, I would either define and explain *"Methods like HOG, SIFT, and Kalman filters"* or switch to a more general language. I don't think your audience will be familiar with these techniques and how they relate to the overall goals of understanding image disturbances. A good piece of advice I got in my scientific career is that if you don't want to explain something (or can't), don't include it in your paper/talk :)
- Again, everything in *"Synthesising an Intelligent System"* I think would be well served by a schematic

- *"After recent development in computer vision, new techniques like deep learning are much superior compared to their ancestors."* — this is a strong scientific claim, that I think you should have a citation for
- It'd be nice to have figure 1 closer to the point you are actually discussing (this is true for all the figures). Also, all your figures need more detailed captions. You should keep the title you have, but add an explanation of the image content. In general, a good rule is that the figure and its caption should be able to give enough information that someone could just look at the figures and understand their point
- You say an advantage of a CNN is robustness in adversarial attacks. What does this mean in the context of self-driving cars? If there isn't really relevance, maybe remove or cut down this section "

Reply: (done and implemented) Schematic has been added, reduction of repetitive sections has been made, definitions of terms and rearrangement of images is done.

Comment 4:" Computer Vision Tasks within Autonomous Systems:

- I'm actually not sure the section "*Classical versus Deep Learning Approaches"* is needed. You introduce a lot of concepts you don't necessarily explain in detail, but I actually think it's not really required for your explicit goal of comparing the interpretability and performance of modern image processing algorithms. At this point, you've already introduced me to CNNs and ViT
- SImilarly, *"Relevance to interpretability"* contains a lot of information presented in your introduction that seems out of place
- Overall, I'd remove this section completely and just add the information to other appropriate sections. For example, consider adding the information about interpretability to *"Synthesising an Intelligent System"* as part of the software
- For figure 1, it's essential to tell me what the heatmaps actually mean when you reference this paper. The audience probably isn't going to be familiar with machine learning, so don't be afraid to feel like you are over explaining "

Reply: (done and incorporated) removal of the section and condensation of repetitive sentences has been implemented.

Comment 5:" **Foundations of AI in Visual Perception Models :**

- Why aren't there citations really in this section? Since I'm sure you're getting this information somewhere, you need to add citations
- *"Due to ViTs' recent growth, they are able to handle and provide significantly more accurate answers for complex scenes than the older CNN models."* - this is a very strong claim. Maybe it's not appropriate for this section of the paper where you are just explaining the overall architecture (and isn't evaluating this kind of the

whole point of the paper?) Otherwise, needs a more thorough discussion and citation

- This section veers a little bit away from your main paper goals and just provides background information. I think because of that, the section could be condensed into one paragraph. Instead, I'd keep things tightly connected to your overall research goal as much as possible, or else the reader becomes a little fatigued "

Reply: (done and incorporated) this section has been dispersed and reduced significantly to satisfy the feedback.

Comment 6:" **Structural Comparison: CNNs and ViTs:**

- This section is overall nice and very important to your paper. As a reader, I kind of felt like "finally, we're getting to the point." It makes me wonder if you should save some of the information you have about CNNs and VIT in the *"Computer Vision and Autonomous Vehicles"* for this section and just shorten the earlier sections. For example, why not just have the figure 1 heatmap interpretability point here. It seems like a more appropriate place.
    - For example, the information on page 8 starting with "*Further research into this drawback has…* "would make more sense in this section to me since you are comparing performance
    - The *"Relevance to interpretability"* could also be condensed and put in the "*Interpretability Differences"* section. Maybe for the introductory material, just saying that it is important for models to be interpretable is enough
- You don't need the section starting with *"This section provides an overview"* . Maybe just one sentence to act as a transition "

Reply: related information from other sections have been added, images have been re-organized and explanations have been incorporated.

Comment 7:" **Safety Risks in AI-Driven AVs:**

- Did you make figure 4 yourself? If so, I'd actually say you require a short methods section, because it is unclear where the data is from and how the number of crashes is calculated. It seems to me that the number is lower than I would expect, but maybe you are focusing on accidents where the AI system was directly responsible for the crash. If so, a robust description of your data is needed
- Relatedly, are your results in figure 4 consistent with what other papers find on crash data? Contextualizing your results would help strengthen your point
- *"Risk factors like adverse weather and low-light contrast demonstrated instability in the model"* — were any of these crashes explicitly caused by these things? If so, state this

- To me, one crash a year is very safe and your point about one pedestrian fatality seems very very small in the context of car fatalities overall. It doesn't align with your conclusion that "*It developed fear within the AV space."*- instead I think you need citations of maybe how regulations were developed in response, or how it tangibly affected self-driving research. "

Reply: (done and incorporated) short method section is provided and explanation of the graph and impact has been discussed and implemented.

Comment 8:" **Types of Noise and Common Techniques for Noise Mitigation:**

- These sections are very good
- My only critique here is maybe to relate more directly your stated research focus on ViTs and CNNs. Maybe discussing how they would fit into the process of training and deploying these models. A diagram could be nice"

Reply: (Done and Incorporated) diagram and explanation has been added.

Comment 9: "**Model Strengths Across Perturbation Scenarios:**

- For this section, how are you deriving the information in table 1? Is it based on literature? If so, you should add citations to the table (and maybe a short description of how you extracted this information in a methods section) "

Reply: (Done) citations have been provided.

Comment 9: "**Observation and Analysis:**

- You need to explain the COCO dataset and what it entails
- Are you deploying models and doing empirical testing in this section? It is very unclear to me, so I think a more thorough explanation of what is the "secondary analysis" you are undertaking is warranted. If you are doing model experiments, you absolutely require a methods section. I would honestly cut a lot of the background if you are short on space and focus on explaining this, as it is absolutely crucial to your results
- Examples of things that need to be explained better:
    - How did you choose the models that are tested? At this point, this is the first I am hearing about things like Faster-R CNN or DETR. How does this relate to your CNN vs ViT focus? How do the models differ? I have no point of reference as the reader to evaluate your results. Even though you have some information about them in table 3, this is really late in your paper
    - Why mAP as the outcome?
    - How realistic are these perturbations?
    - What are the outcomes you are trying to predict? How realistic are these?

    - What image perturbations are you doing specifically? Are you doing just one type or multiple?
    - Do the models do any of the *Mitigation and Robustness Strategies* you carefully described earlier? This could provide context into your results for the reader (it wouldn't be that surprising that a model not trained on perturbed images would have reduced performance when tested on these types of images)
    - Are any of these models actually deployed in real self-driving cars? This would help to contextualize the magnitude of the results-is it really bad that the mAP decreases so much, or is this more of a research algorithm that isn't designed for deployment
- The advanced perturbation analysis is nice
- I'd include the equations in a methods section instead. You also need to define the parameters in the methods section and provide a caption for the equations
- Have any other papers used this benchmark? Are your results consistent with these papers? Why or why not? Engaging with the literature in this way could help add nuance to your results
- Table 3 states "*Reliability distributions for CNNs versus ViTs."* but you actually don't give me a lot of context for which algorithms are using CNN or ViT. I really like this section, but there are so many rich results, it loses the plot a little in respect to the stated goal in the introduction- explicitly comparing the VIT and CNN. Making this comparison more explicit in this table and in the whole section would really help your readers
- Overall, this is a cool section, but it felt a little secondary to other parts of the paper. I think you should be really explicit about how it connects to the literature that you spend so much time reviewing and in the conclusion section, why this analysis matter "

Reply: (done and implemented) COCO dataset's content has been explained, a method section contains the algorithms, the selection of paper, and training or evaluation of models has been described.

Comment 10: "**Discussion:**

- I'm kind of confused about the goal of this section, since you also have a conclusions section. To me, your whole paper is kind of a discussion. I expected this section to be more about the contextualization of your analysis results in the literature, so I was surprised to see it repeating some information you said earlier. Maybe you could combine the conclusions into one cohesive section that addresses the conclusions of your literature review and how your analysis results fit into the literature more broadly?

- You are missing citations throughout; it's okay to repeat citations you have already used, but things like "*CNNs have greater potential to excel in local texture recognition"* aren't directly interrogated by your analysis so need to have a citation

Future directions:

- I think also could benefit from some citations
- Since you did analysis, maybe you also want to state the future directions of analyses like yours?
- I honestly think this could be condensed into one discussion/conclusions section that goes in this paragraph order:
    - Overall results
    - Contextualization of results in literature
    - Limitations and future directions
    - Final thoughts and conclusions "
      Reply: (done and implemented)

Warm regards,
[Author name redacted]

Overall comments

Overall, great work at addressing the comments! The paper is much improved in my opinion

- Originality & Significance – This paper is very original and robust. I
- Clarity & Structure – The clarity and structure is much improved by adding a methods section and shortening unnecessary sections. Good work!
- Use of Evidence & Research Methods – I am happy to see a methods section. I'd recommend some minor improvements to make it slightly more clear, like adding equation numbers, and some subheaders that point to the different types of analysis
- Engagement with Literature – This was overall good. I still think the engagement with the literature around the author's specific benchmarking analysis could be improved, but I really feel like only a few sentences in the discussion would be needed to just explicitly connect what was found in this study with the broader field and would be very minor overall
- Grammar & Language – Good!

Decision: Accept with minor revisions

Detailed comments:

Abstract:
Much clearer. I think you even undersell a little by saying you do “secondary data analysis”- I think you could say that you perform robustness analysis using COCO and it would be clear immediately what you did

Introduction:
Really good! Great improvements. Minor feedback is to make the last sentence of the introduction a little clearer on what exactly you did in this analysis

Computer Vision and Autonomous Vehicles:
The addition of the new figure 1 is very helpful. Thank you for including. The length cuts also substantially improve the focus of the paper.

Computer Vision Tasks within Autonomous Systems:
Minor comment but there’s just a large gap after page 12.

Safety Risks in AI-Driven AVs:
I think your discussion of the accidents is much more robust now. I’d just ask for citations about the new safety regulations and temporary testing suspensions

Methods:
Very happy to see this section included, and would just say you should probably include the following to make it more robust:

- A short description of your analysis of the accident data
- Equation numbers
- A description what software you used to calculate the relevant statistics and plots
- Feel free to use subheaders

Also it is standard in scientific publishing for the methods to either be at the end or the beginning of the paper, not in the middle. I’d recommend the end for your paper.

Key observations and analysis:
Minor comment but it is hard to read figure 12 and 13. They’re a little blurry

Discussion and conclusion:
This is much more coherent. I’d recommend perhaps breaking the text up into multiple paragraphs, however? I’d also love to see more citations or sentences that directly compare your findings to what other people in the field have found.