Adopting Explainable AI to Autonomous Driving Agents

Amel Nestor Docena, Amittai J. Wekesa, Andrew P. Hederman,
Hannah M. Brookes, Kevin Lin, Mingi Jeong, Parth Dhanotra, and Phuc Tran

Keywords: Explainable AI, transparency, decision-making, autonomous agent

1 Introduction

Advancements in artificial intelligence (AI) have made tremendous technological advancements in the areas of natural language processing, bioinformatics, and autonomous vehicles (AVs). The ability of these AI agents to correctly perform tasks has increased research interest in these areas. While overall task performance has grown exponentially, the ability to understand the decisions within the algorithm that lead to the result remains unclear: machine learning or deep learning networks continue to be plagued by the **"black box"** problem due to the complexity of understanding their inner workings (Fig. 1). We stress that the widespread implementation and acceptance of AI agents are highly dependent on their trustworthiness, and the ability of users and developers to interpret their decision-making processes. And so, understanding why these AI agents are making specific decisions is crucial.

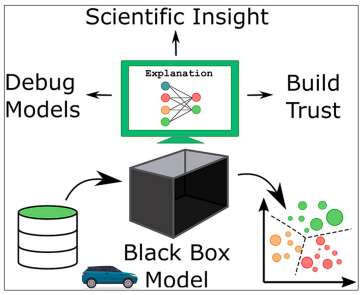


Fig. 1: Necessity of AI explainability in autonomous agent – modified figure from [1].

We thus analyze the state-of-the-art approaches and propose an interpretable system that embeds **"explanations"** on why decisions are made (or what led to such decisions), to bridge any gaps. For example, this system would provide critical information on how a self-driving car decides on risky situations such as near collision with an obstacle, enlightening "why?" and motivating "what if?" for better decisions. By serving as a *review article*, our proposed approach focuses on the scope of autonomous driving and aims to make the following contributions:

- identify techniques of explainable AI (XAI) in AV domain
- conduct experimental analysis on real-world implementations
- propose an XAI framework for autonomous self-driving agents
- design and implement a system for scaling these techniques

The proposed work seeks to provide insights with a broad range of applicability, from increasing trustworthiness and understandability in the eyes of lay users to providing a better framework for research. The application of our study can be further extended to general autonomous decision-making systems and even human-operated systems.

2 The Autonomous Vehicles Paradigm

The problem of explainability with respect to autonomous vehicles can generally be separated into four categories: perception, localization, planning, and system management. [2]. In the following sections, we will shed some light on the modern XAI research landscape by highlighting key findings from each domain.

2.1 Perception

Perception is the first step in the XAI paradigm for autonomous vehicles. Perception models take in raw data, such as sensor data from cameras, LiDAR, RADAR, GPS, etc., and translate the input into a computer-based representation of the real world, mainly through object detection [2]. Many models that are used for perception are "black-box" models, as described above, and may prompt the end-user (driver) to raise the important questions of explainability: Why did the model make the given prediction? What regions of the image (or other input data) were most important in making that given decision?

To answer these questions, a variety of alogrithms exist. Such algorithms can largely be grouped into three categories, gradient-based methods (such as class saliency maps, Grad-CAM, DeConvNet, and guided backprop), activation-based methods (such as class activation mapping (CAM), attention branch networks (ABN), and layer-wise relevance propagation (LRP)), and finally occlusion or perturbation methods (such as algorithms like LIME or SHAP) [3]. The goal of all such algorithms are to highlight what features, which in the class of

autonomous vehicle perception is often pixels of an image, were most important in the model's decision making process. These are often visualized by overlaying a heatmap over the original image to emphasize the most meaningful parts of the image.

Grad-CAM, one of the most commonly used XAI methods, utilizes gradient-weighted class activation mapping (See Fig. 2 for an example). It employs backpropagation to compute gradients at the final convolutional layer, which are then used to weight feature activation maps. This process generates a localization map, highlighting the crucial regions in the input image for determining the target class (see [4] for calculation details). Unlike CAM, which is limited to networks without fully connected layers, Grad-CAM functions without altering the model architecture. By enhancing the explainability of black-box algorithms, Grad-CAM's visualizations can serve as an interface between the computer model and the driver of an autonomous vehicle, thereby enhancing the trustworthiness of the AI agent.

2.2 Localization

Localization is complementary to perception in XAI for Automated Vehicles. Precise and robust localization is critical to AV operations as part of the XAI paradigm. Error margins of sensor data and relative decision making power of each localization instrument must be communicated to stakeholders in XAI. The state-of-the-art localization technique is Simultaneous Localization and Mapping (SLAM), a technique to simultaneously constructing a map of the surrounding environment and estimating the sensor motion through space [5]. Since it's introduction, SLAM has been iteratively developed into it's modern day deployment in AVs, a Visual-LiDAR based SLAM with sensor fusion. SLAM was initially represented with a Bayesian approach with a Maximum a Posteriori (MAP) estimation that would use landmarks in the enviornment to determine the pose of the AV. The Extended Kalman Filter (EKF) was one such solution to SLAM MAP, guaranteeing convergence, but being very sensitive to data association errors with compounded localization errors from state to state [6]. The modern, non-probabilistic, approach to SLAM is a graph based approach proposed in 1997 [7]. The raw sensor measurements are abstracted by edges in a graph, which represent probability distributions over transitions between to possible location states, of the AV, represented by two nodes. The current problem lies in sensor fusion and reconciling the input data from multiple data sources and SLAM techniques, some like V-SLAM which generate feature maps, and others such as LiDAR-SLAM which generate 3D scene maps [8]. For XAI, the use of Shapley Additive Explination (SHAP) values can determine what set of sensor data is emphasized for AV decision making. SHAP is an explainability technique that assigns each feature an importance value for a particular prediction by considering the power set of all features (LiDAR, GNSS, Visual, IMU, etc.) and compares model outputs trained with seach subset of features to generate the Shapley value [9]. Shapley values combined with communication of SLAM localization error and other anomalous sensor error is paramount to XAI in the domain of Autonomous vehicles.

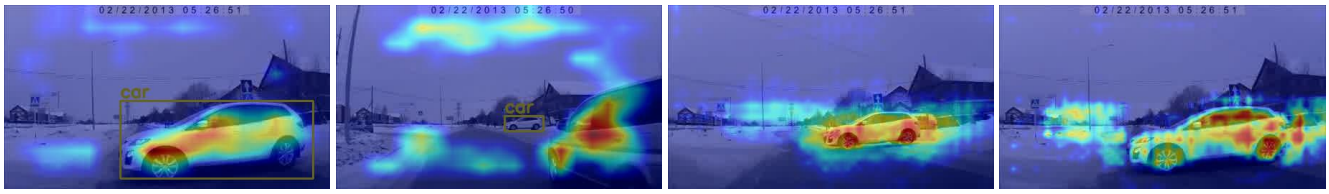
2.3 Planning

In autonomous vehicle and robotics research, planning has been shown importance in delivering safer and more accurate deep learning models. In broad terms planning is any computational process that uses a model to improve its policy by generating future trajectories. Planners are general purpose solvers that commonly utilize search algorithms. Previous work on incorporating planning into deep learning has focused on utilizing reinforcement algorithms in game environments where the reward function can be clearly specified. An early example was the DYNA-Q algorithm that used a reinforcement algorithm approach with a planner to solve a maze. Although simple in idea the results clearly show how a planner can achieve the goal significantly faster than a model without planning. This was also shown for game play where the models were trained on pixel data. Pixel data is extremely relevant for self-driving where the planner will take input data from perception which is commonly in the form of images or videos that will be broken down into pixels. Planning in self-driving is far more complex than a stand-alone search problem with planners needed for motion, behavior, and mission. Research has been done to form novel planning architecture that integrates these different aspects in a more parallel manner. Taking all the inputs from perception and localization the planner uses a predicted cost function to then chose the next best course of action that optimizes parameters such as progress, comfort, safety, and fuel consumption. In addition to the complexity of planning in self-driving there is also an urgent need for explainability of these algorithms which currently does not exist.

2.4 System Management

In XAI with Autonomous Vehicles, the final problem once Perception, Localization, and Planning data is recorded on the AV is an effective method to extract data such that the data is accessible for future XAI operations. There exists two halves to this System Management problem. The first being maintaining data authenticity such that we can ensure data is not modified by an adversary. The second problem,

being maintaining data integrity such that we can ensure data is not corrupted unintentionally. A Blockchain Box Event Recording System is one such solution in AVs for data authenticity. In this system, we can find the root error in a XAI liability paradigm through accident forensics by Proof of Event with Dynamic Federation Consensus [10]. When an event occurs, the recorded data from involved AVs are broadcasted to a local community within Dedicated short-range communication (DSRC) range. This ”community” of AVs broadcasts accident data to the larger AV network, where a random group of AVs are chosen as a ”federation”. The ”federation” is responsible for verifying data integrity and signatures across broadcasters, then providing their own signatures on the accident data as well. A single AV from the ”federation” is chosen as the ”lead verifier”, who then writes the accident data to the blockchain[10]. Any adversary that would attack this network would need to either compromise a majority of the AV ”community” in real time, or identify the ”federation” in real time and compromise a majority of the ”federation”. A Smart Black Block System using deterministic mealy machines (DMM) and local buffer optimizations (LBO) can help us maintain selective data integrity[11]. One difficulty with data extration in XAI is that onboard data can be produced at rates up to 1gb/second. However, much of the data produced is of minimal interest and relation to a crash event. A proposed solution is to use a DMM to group high value data into buffers based on new data values, data similarity, and current buffer size. LBOs help determine an optimal compression value for the current buffer. When on board storage is full, a priority queue discards lower value buffer in favor of high value buffers[11].



(a) (b) (c) (d)

Fig. 2: Testing of Class Action Maps (a-b) with a backbone RCNN model; (c-d) Semantic Segmentation model on accident number 53.

3 Towards Explainable Autonomous Driving

In this section, we conduct research in order to identify the most salient avenues of exploration for future work in the self-driving XAI domain. We arrive at the following list of limitations currently pervading the field: (1) models lack transparency and intuitive explainability at a level that resonates with laypeople, which is crucial to adoption in practice; (2) powerful XAI algorithms already exist, but are difficult for ordinary people to understand (3) these algorithms often lack the infrastructure necessary to scale them to state-of-the-art, large-scale models for self-driving.

3.1 Experimental analysis of XAI on real-world implementations

Algorithms and Dataset: We conducted an experimental analysis of open-sourced, academic XAI works for benchmarking in AVs. For consistent testing in common data, we used *CarCrash Dataset* (CCD) [12] to test heterogenous XAI algorithms and identify what the algorithms can tell in accident scenarios critical for explainability. The dataset contains (1) binlabels indicating where an accident frame is; (2) timing such as day and night; (3) weather such as normal, snowy, and rainy; and (4) egoinvolve for whether the egovehicle is directly involved in the accident. We extracted only data where **egoinvolve** is true in order to have ego-centric explainability.

Fig. 3: Testing of Deep Feature Factorization on accident number 56.

Class Activation Maps for Object Detection with Faster RCNN [13]: For our first experiment, we completed a qualitative Class Activation Map on *CarCash Dataset's* accident number 53. Within the CAM, we focus on the class scores of the objects detected in our video since these are the only class scores given by an RCNN model. In our test, the best result was shown when the entire car was in the middle of

the frame, and the boundary box was easily drawn (See Fig. 2 (a-b)). There were no other object within the frame. On the other side, a poor qualitative result occurred when the frame presents multiple objects of the same class. While the boundary boxes focused on the smaller upcoming car, the heat map showed that the nearby car's pixels contribute the most to the class *Car*. However, since the frame cuts off the entire shape of the nearby car, no boundary boxes were drawn. The FasterRCNN is limited to simple pre-trained images and might not fit a video with fast sequential events.

Class Activation Maps for Semantic Segmentation [14]: With a Semantic Segmentation model, each pixel is given a score value for its best-predicted class for that pixel. By nature of the model, the pixels surrounding each pixel influence that pixel's predicted class. For a Class Activation Map, a semantic segmentation model can be exploited to sum up the scores of all predicted pixels of that class [14]. When we focus on the gradients with strictly the predicted pixels, there are more granular details, and the heat map is not restrictive to the boundary boxes like the RCNN backbone example (See Fig. 2 (c-d)). Despite the heat map's qualitative improvement, the strongest heat map comes from a standard picture, in which the car is in the middle of the frame. As the car comes closer to the egocentric view, the heat map significantly becomes less centralized.

Fig. 4: Testing of *PilotNet* on the CCD dataset: (left) steering control explainability, (right) saliency map for steering.

Deep Feature Factorization for Better Model Explainability [15]: Deep Feature Factorization (DFF) is a visual-focused XAI model that allows us to restrict images to abstract objects. It enables unsupervised real-time image processing in autonomous vehicles and provides insights into the tendencies of Deep Learning models. As shown in Fig. 3, DFF visualizes clusters that carry significant implications, even if they are not dominant in the entire image. In our research on CarCrash footage, DFF effectively segments important areas without specific output labels. Expanding DFF's capabilities with cross-modal analysis can enhance its results in different situations.

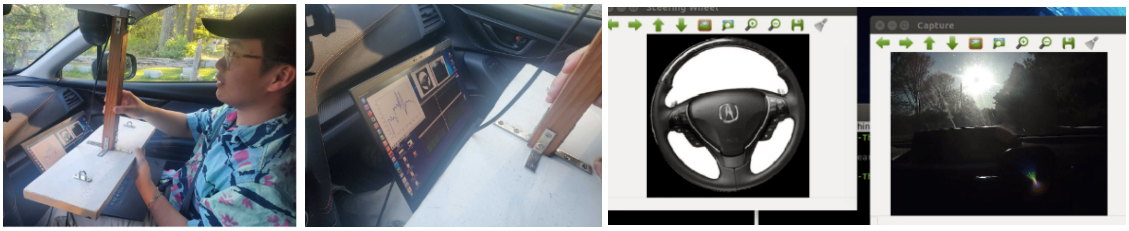
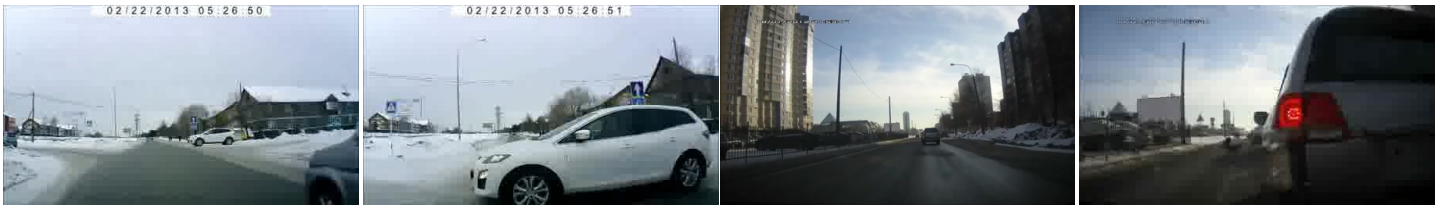


Fig. 5: Real-time online experiment: (left) system configuration. (right) a challenging scenario with sun glare.

Explainability on Steering [16], [17]: *PilotNet* was developed by NVIDIA and trained based on images with time-synchronous steering command. The architecture is simple by consisting of several convolutional layers and fully connected layers in the final steps such that the algorithm returns the steering command given an image. (1) We first tested the algorithm with the recorded CCD dataset about how well the algorithm performs by extracting the control history. Fig. 4 (left) shows the algorithm is good at explaining the cause of the accident by showing the lane crossing as a faulty maneuvering of the ego vehicle. (2) We also investigated why the steering command was derived from the proposed network by Grad-Cam Fig. 4 (right)). (3) Finally, we tested the proposed algorithm by a modification to adopt online video input (OpenCV) to test with online and real-time capability (Fig. 5 (left)). During the experiment, we also encountered challenging lighting conditions – Fig. 5 (right) – that hinder the running of the XAI algorithm, which is also common to the general perception algorithm in the self-driving car domain. Such a result indicates that we need more redundancy in case of the missing capability of a single XAI algorithm.

Explainability on Action with Natural Language [18]: ADAPT (Action-aware Driving cAPtion Transformer) proposes a transformer-based architecture that offers user-friendly natural language narrations and reasoning for each decision made during autonomous vehicular control and action. ADAPT simultaneously trains the driving caption task and the vehicular control prediction task by utilizing a shared video representation which is trained on the BDD-X (Berkeley DeepDrive eXplanation) dataset [19]. As shown in Fig. 6, we tested ADAPT on the CCD dataset for real-world applications by modifying the input capabilities. Fig. 6 (a) shows that the correct prediction about the car's action prior to the accident. Fig. 6 (b) shows the literal action of the ego vehicle with respect to the traffic moving together, based on a myopic view. However, the action does not explain the holistic context of the corresponding crash accident.



N: the car slows to a stop. **R**: there is a car in front of it. **N**: car is driving forward. **R**: traffic moves at a steady speed.

Fig. 6: Testing of ADAPT on CCD dataset where *N* is anarration and *R* is a reasoning.

Domain Shift We also tested *PilotNet* with a question on "how the XAI algorithm will behave under different domain?". For this domain shift problem, we used maritime domain data collected by Dartmouth Robotics (Fig. 7). The data format was in `ROSBAG` with 70*GB* and 2.5 hours. We processed the data for time-synchronous image (RGB camera) and steering command (IMU). The result shows that (1) originally trained model under the car domain does not work well; and (2) newly trained model also showed a poor performance (marine domain loss: about 3.0 MSE vs. car domain loss: less than 1.0 MSE). For this analysis, we looked at the explainability by Grad-cam again such that we are able to find the main cause could be not explicit features having a causal relationship with a steering command in water. The result justifies our assumption that the water domain does not have a clear lane boundary (Fig. 7 – mid, right).

Fig. 7: Domain shift experiment. Data collected at Busan, Korea by Dartmouth Robotics Lab. (left): navigation route; (mid, right) saliency map at different frames.

3.2 XAI on Reinforcement Learning

In the pipeline of autonomous self-driving, the predictions of perception models and localization, (e.g., identifying traffic signs, obstacle detection, etc.), are used to define the present state of the agent and its environment. These states are then fed to a policy learning module to plan the "optimal" set of actions to take, (i.e., the policy); here we explore reinforcement learning as the module. What are some state-of-the art XAI techniques on RL? And how can they be relevant in explaining autonomous self-driving vehicles' policy?

In their work, Lin, et al. [20], introduce learning action-values that embed human-understandable features (or properties) of expected futures; (example of a feature could be, say in self-driving car, its velocity). Recall that in learning the policy for RL, we use the Bellman equation that measures the return of taking an action, a from current state, s: $Q(s,a) = R(s,a) + \gamma \Sigma_{s'} T(s,a,s') Q(s', \pi(s'))$; i.e., the current return plus expected future return behaving optimally after taking that action. What [20] propose is to devise a Bellman equation for "user-provided" features, F as well: $Q_F(s,a) = F(s,a) + \gamma \Sigma_{s'} T(s,a,s') Q_F(s', \pi(s'))$. They coin this as *generalized value function*. What GVF captures is the current feature measure by taking action, a from current state, s plus the expected future feature measure behaving optimally. Thus, we are able to capture meaningful properties of a policy's future trajectories that are represented by these features.

To test this technique out, we run an experiment on an autonomous lunar lander that uses RL for its policy in landing to a goal in the moon safely, (Fig. 8 (left)). The features we provide to understand the policy trajectory are distance to goal, velocity, tilt-angle, right (and

left) landing leg in goal position, main engine use, side engine use, and whether a safe landing. We find that learning GVFs alongside the policy learning helps us (as the user) understand the trajectory of said features that to us are meaningful as the lander follows its learned optimal policy. We find that other SoTA XAI on RL interprets the trajectory of learned policies as well, [21]. These techniques are relevant in autonomous self-driving as they can aid users understand properties of the agent's policy.

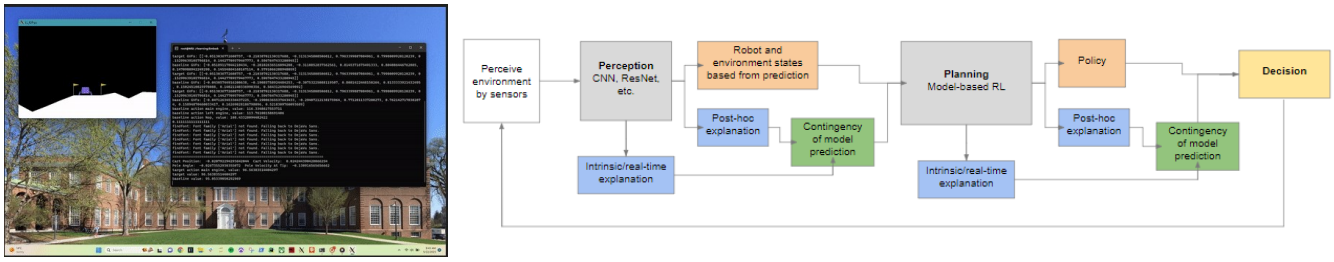


Fig. 8: (left) Self-driving lunar lander using GVFs experiment for RL XAI; (right) Proposed XAI Framework for self-driving agents.

3.3 Proposed XAI Framework for Autonomous Self-Driving (An Idea):

To put all these XAI pieces together, Atakishiyev, et al. [22] propose an XAI framework for autonomous self-driving. We build upon that what we have learned so far, (see Fig. 8 (right)). The idea is that, in the pipeline, we re-use the intrinsic and post-hoc explanations, (blue nodes in the figure), that become available to users as by-product of the model predictions. We develop a contingency model (green nodes) on how reliable the model predictions are, feeding this contingency as extra input alongside the model predictions for the next step in the pipeline.

3.4 Scaling Explainability

Lastly, we consider implementational issues surrounding modern explainability techniques. Modern, large-scale models require the capability to arbitrarily intervene on intermediary activations in order to interpet them, as the bulk of their capabilities stem from the hierarchical composition of intermediary components [23], [24]. Dominant machine learning frameworks like PyTorch approach neural networks from an object-oriented perspective, in which networks are implemented as arrangements of smaller modules [25]. This, in turn, is reflected in the implementation of systems for interacting with model activations. For instance, research groups like Anthropic employ closed-source tooling inspired by PyTorch's 'hook system', in which individual modules are assigned functions which can collect and manipulate activations [26]. However, with the advent of new interpretability techniques that manipulate models in increasingly complex, unconvential ways [27], as well as the sheer scale of modern AI systems, there is a need for a more flexible system that can intervene on model activations without concern for the particularities of its architecture or implementation. As a proof-of-concept, we implemented a system that generates a computational graph representing a model, and allows users to match sub-graphs to arbitrary functions representing model components. In this manner, users can manipulate activations without interfering with model internals - a demonstration can be found here.

4 Conclusion and Future Works

Our work serves as a review article on explainable AI (XAI) in autonomous vehicles (AVs) from both a post-hoc and model perspective. Our XAI paradigm combines elements of perception, localization, planning, and system management as an end-to-end pathway for navigating explainability in AVs. The specific regions attended to by an AV model can provide insights into explainability.

After conducting a comprehensive review of general XAI models for AVs, we evaluated the advantages and disadvantages of each model using the CarCash Dataset [12]. Our zero-shot results provided valuable insights for each model. In particular, we found that combining a strong autoencoder with Deep Feature Factorization led to the formation of explainable abstracted clusters of objects. Our experiments in real-time testing and domain shifts revealed that the model had limited contextual knowledge about water navigation rules, even after training. To address this, we propose an XAI reinforcement learning (RL) framework that incorporates natural language processing and strong latent features, drawing inspiration from [21]. Additionally, we present a proof of concept for using modular functional blocks as deep learning models, aiming to move away from computationally heavy object-oriented programming.

Looking ahead, we envision task-specific methodologies for different domains and analyses, such as post-accident or real-time scenarios. As autonomous driving becomes more prevalent in our daily lives, continuous human survey research is crucial. It is worth noting that XAI research does not rely on a one-size-fits-all method for all humans, and ongoing research is necessary to meet the evolving needs of the field.

References

[1] F. Oviedo, J. L. Ferres, T. Buonassisi, and K. T. Butler, "Interpretable and explainable machine learning for materials science and chemistry", *Accounts of Materials Research*, vol. 3, no. 6, pp. 597–607, 2022.

[2] D. Omeiza, H. Webb, M. Jirotka, and L. Kunze, "Explanations in autonomous driving: A survey", *IEEE Transactions on Intelligent Transportation Systems*, pp. 10 142–10 162, 2022.

[3] C. Molnar, *Interpretable Machine Learning, A Guide for Making Black Box Models Explainable*, 2nd ed. 2022.

[4] R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization", *IEEE International Conference on Computer Vision*, 2017.

[5] R. Chatila and J. Laumond, "Position referencing and consistent world modeling for mobile robots", in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, 1985, pp. 138–145.

[6] J. Leonard and H. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons", *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 376–382, 1991.

[7] F. Lu and E. Milios, "Globally Consistent Range Scan Alignment for Environment Mapping", *Autonomous Robots*, vol. 4, no. 4, pp. 333–349, Oct. 1997.

[8] G. Jiang, L. Yin, S. Jin, C. Tian, X. Ma, and Y. Ou, "A simultaneous localization and mapping (slam) framework for 2.5d map building based on low-cost lidar and vision fusion", *Applied Sciences*, vol. 9, no. 10, p. 2105, May 2019.

[9] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions", in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, *et al.*, Eds., vol. 30, Curran Associates, Inc., 2017.

[10] H. Guo, E. Meamari, and C.-C. Shen, "Blockchain-inspired event recording system for autonomous vehicles", in *2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN)*, 2018, pp. 218–222.

[11] R. Feng, Y. Yao, and E. Atkins, "Smart black box 2.0: Efficient high-bandwidth driving data collection based on video anomalies", *Algorithms*, vol. 14, no. 2, p. 57, Feb. 2021.

[12] W. Bao, Q. Yu, and Y. Kong, "Uncertainty-based traffic accident anticipation with spatio-temporal relational learning", in *ACM Multimedia Conference*, May 2020.

[13] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization", in *IEEE CVPR*, 2016.

[14] K. Vinogradova, A. Dibrov, and G. Myers, "Towards interpretable semantic segmentation via gradient-weighted class activation mapping", *CoRR*, vol. abs/2002.11434, 2020.

[15] E. Collins, R. Achanta, and S. Susstrunk, "Deep feature factorization for concept discovery", in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 336–352.

[16] M. Bojarski, D. Del Testa, D. Dworakowski, *et al.*, "End to end learning for self-driving cars", *arXiv preprint arXiv:1604.07316*, 2016.

[17] M. Bojarski, P. Yeres, A. Choromanska, *et al.*, "Explaining how a deep neural network trained with end-to-end learning steers a car", *arXiv preprint arXiv:1704.07911*, 2017.

[18] B. Jin, X. Liu, Y. Zheng, *et al.*, "Adapt: Action-aware driving caption transformer", *arXiv preprint arXiv:2302.00673*, 2023.

[19] J. Kim, A. Rohrbach, T. Darrell, J. Canny, and Z. Akata, "Textual explanations for self-driving vehicles", in *ECCV 2018*, 2018, pp. 577–593.

[20] Z. Lin, K. Lam, and A. Fern, "Contrastive explanations for reinforcement learning via embedded self predictions", in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, 2021.

[21] J. Chen, S. E. Li, and M. Tomizuka, "Interpretable end-to-end urban autonomous driving with latent deep reinforcement learning", *IEEE Trans. Intell. Transp. Syst.*, pp. 5068–5078, 2022.

[22] S. Atakishiyev, M. Salameh, H. Yao, and R. Goebel, "Explainable artificial intelligence for autonomous driving: A comprehensive overview and field guide for future research directions", *CoRR*, vol. abs/2112.11561, 2021.

[23] C. Olsson, N. Elhage, and N. Nanda, "In-context learning and induction heads", 2022.

[24] C. Olsson, N. Elhage, and N. Nanda, "A mathematical framework for transformer circuits", 2021.

[25] A. e. a. Paszke, "Pytorch: An imperative style, high-performance deep learning library", 2019.

[26] N. Elhage, "Garcon", 2021.

[27] L. Chan, G.-A. Adrià, N. Goldowsky-Dill, *et al.*, "Rigorously testing interpretability hypotheses using causal scrubbing", 2022.