Practical Self-Driving Cars: Survey of the State-of-the-Art

Debanjan Saha†* Shuvodeep De‡*

†P360, Mumbai, Maharashtra, India debanjan.aiml@gmail.com
‡Virginia Tech, Blacksburg, VA, USA shuvode@vt.edu

***Abstract*—Self-Driving Vehicles or Autonomous Driving (AD) have emerged as the prime field of research in Artificial Intelligence and Machine Learning of late. The indicated market share of existing vehicles might be supplanted by these self-driving vehicles within the next few decades. While AD may appear to be relatively easy, in fact, it is quite the contrary owing to involvement and coordination amongst various kinds of systems. Numerous research studies are being conducted at various stages of these AD systems. While some find the various stages of the AD Pipeline beneficial, others tend to rely on Computer Vision mostly. This paper attempts to summarise the recent developments in Autonomous Vehicle architecture. Although some people might seem to be sceptical about the pragmatic use of AD as an alternative to existing vehicles, the plethora of research and experiments being conducted suggests the opposite. Indeed, there are many challenges to implementing AD in the real world, but significant progress made in the last couple of years indicates general acceptance of AD in upcoming years.**

***Index Terms*—Self-Driving Vehicle, Autonomous Driving, Imitation Learning, Semantic Segmentation, AD Pipeline, Computer Vision, Deep Learning.**

I. Introduction

Self-driving cars, or autonomous driving (AD), are a concrete application of robotics and artificial intelligence. As its name suggests, a self-driving car is capable of driving on its own without any human intervention. For that purpose, it must perceive its surroundings and then move accordingly. Self-driving cars are being equipped with a diversity of sensors, cameras as well as actuators for scene perception. High-definition maps and protocols for computer vision, localization, sensor fusion, prediction, planning, and controls are the accessories of a self-driving car. Advanced and well-matured concepts of artificial intelligence, machine learning, big-data handling and control systems are the real essence behind self-driving technology. Presently, self-driving cars are certainly a hot topic in research and developments throughout the world. Innovations and technical advancements are being proposed in this field incessantly. While there are a few papers which have surveyed various deep learning architectures beneficial for AD, none of them talks extensively about the ground reality of the adoption of these techniques and the consequences it bears. This paper tries to address these issues while talking about realistic AD vehicles.

There exist five stages of vehicle automation. The zero automation stage is a pure manual driving system that depends on human decisions only. The first automation stage, also called "Driver Support," includes some intelligent features such as parking sensors. The second automation stage incorporates restricted automatic cruise control such as the "Lane-keeping System", but the human driver has main control over the vehicle's operation. The third stage of automation, also called "Conditional Automation", enables human drivers to switch from manual driving to automatic driving for a longer time period (such as highway driving), but a human interface is required occasionally. The fourth stage is "High-level Automation", where no human interface is provided. The fifth stage is "Fully Autonomous", where no human interface or even human is required [1].

A. Advantages over conventional vehicles

Self-driving cars offer the following opportunities and advantages over conventional ways of driving.

- Fewer accidents and a safer driving environment: Each year, approximately 1.25 million fatal vehicle accidents are recorded worldwide. Every year, more than forty thousand people die in automobile accidents alone in the United States [2]. The main causes of such accidents are found to be drowsiness, incapacitation, inattention, or intoxication [3]. According to advocates, most traffic accidents (more than 90%) are caused by human errors. Therefore, the utilisation of novel materials like stiffened composite materials [4]–[20] and advanced self-driving cars has the potential to reduce accidents by 90% [21].
- Reduced Congestion and Improved Reliability: As humans have restricted perception and reaction times, they can't efficiently deploy highway capacity. Self-driving cars can improve efficiency and reduce congestion as they are computer-operated and connected. Consequently, the existing highways will be used by more vehicles. When existing roads have more carrying capacity, there will be less pressure to build new roads and expand existing ones to accommodate congestion. Ultimately, the land proportion reserved for roads and parking areas can be deployed for commercial and residential purposes. Moreover, increased penetration of self-driving vehicles into the transport system will assist in reducing traffic delays and vehicle crashes while improving system reliability [2].
- Environmental Benefits: As self-driving cars facilitate a safe and reliable transportation system, Therefore, the ve-

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

hicle's design can also be transformed from safe, protective versions (tank-like) to lighter versions, which would have fewer fuel requirements. Moreover, the industry of electric vehicles (EVs) and other propulsion technologies are also being supported by self-driving concepts. The overall decrease in fuel consumption will ultimately bring the most cherished environmental benefits.

- Mobility for Non-drivers: Self-driving cars provide an opportunity for non-drivers (young, old, impaired, disabled, and people who do not possess a driving license) to have personal mobility [2].
- Reduced Driver Stress: A study is being conducted with 49 participants, which demonstrates that the self-driving environment is less stressful as compared to manual driving [22]. Therefore, self-driving cars may ameliorate the overall well-being of people by reducing driving related workload, associated stress, and normalising their heart rate [2].
- Improved Parking: Self-driving cars are anticipated to decrease parking space significantly (by more than 5.7 billion square metres in the US alone) [2]. There are multiple factors contributing to parking improvements, such as the requirement for open-door space in the parking area for self-parking vehicles, as there is no human driver who needs to get out of the vehicle. Therefore, vehicles can be parked more tightly (almost 15%).
- Value Added Time for Non-driving Activities: Self-driving cars allow people to save time (as much as 50 minutes daily), which they would otherwise spend on driving activities. People can invest this time in working, relaxing, and entertaining themselves.
- Shared Mobility: Nowadays, a new concept, "mobility-on-demand", is trending in mega-cities. Self-driving vehicles are anticipated to be supportive of this trend as self-driving taxis, private vehicle ride-sharing, buses, and trucks would automatically be able to facilitate high-demand routes. Vehicle ownership can be reduced up to 43% while travel per vehicle can be increased up to 75% with this concept of shared mobility.
- Real-Time Situational Awareness: Self-driving cars can deploy real-time traffic data (such as travel time and incident reports) and can be more efficient and sophisticated while performing well-informed navigation system and vehicle routing [2].

B. The intersection between self-driving cars and electric vehicles (EVs)

The concept of self-driving can be applied to EVs as well, but it has its own trade-offs. As self-driving cars introduce a safer and more reliable driving environment, it will encourage a transition from heavier (safety-protective) vehicle designs towards lighter designs. Lighter vehicle designs are more suitable for electric propulsion [2]. On the other hand, self-driving cars must be equipped with myriad of sensors and computers for processing, which consume a lot of energy. While EVs have limited battery range [23]–[25], therefore self-driving EVs would be able to cover only shorter distances. However, experts believe that these hurdles can be overcome with further technical advancements. As certain features of autonomy require more energy than other, these features can be optimized with intelligent software and hardware adjustments. Consequently, travelling range of self-driving EVs can be ameliorated.

C. Why self-driving cars became possible due to the development of Artificial Intelligence (AI)

AI is the real essence behind self-driving cars or autonomous driving. AI is basically imitation of human cognition and intelligence by a machine. Machines that are being programmed with AI can perceive, learn, autocorrect, make the right decisions, and do reasoning [26]. Because a self-driving car has to drive a car like a human does, which involves a lot of complicated tasks such as scene perception, static and dynamic object detection, safe navigation, observing traffic rules, speed control, lane keeping and switching, turning at intersections, and prevention of accidental scenarios. Moreover, the presence of pedestrians and bicycles in overcrowded urban environments adds even more burden to the self-driving mechanism. Complicated and advanced concepts of AI, machine learning, big-data handling, and control methodologies play significant roles behind self-driving technology [27]. AI is being used in many other disciplines related to the AD industry, in addition to the driving mechanism of self-driving cars.The process of car manufacturing can be considered as a puzzle solving problem, which involves accurate fitting of a myriad of components. Car manufacturing companies are widely deploying robotic assistance and AI for precise and meticulous assembling of car constituents. AI is also playing its role in car safety and driver's protection in the form of emergency braking and vehicle control, observing cross-traffic and blind spots, and synchronising with traffic signs. Moreover, AI is being used for well-timed suggestions for predictive and prescriptive maintenance of cars by continuously observing their physical condition. Regulators and insurance companies are also getting advantages from AI ubiquity. AI can accumulate data about a driver's behaviour and risk profile and provide it to the insurance company for an accurate assessment of insurance costs. Furthermore, AI is also facilitating drivers by regulating seat position, mirror, air-conditioner, even music according to their choices.

D. Statistical predictions about expansion of self-driving cars industry in near future

The industry of self-driving cars, or AD, has the potential to transform the entire transportation system in the future, because of its potential benefits over manual driving. According to statistics being collected in 2018, the worldwide market for self-driving cars is anticipated to expand from its current state of $5.6 billion to almost $60 billion by 2030. Moreover, the production of self-driving or robo-cars will also increase at the same pace. According to statistical prediction for 2030, the production levels will be increased up to 800,000 units annually all over the world. Furthermore, many small businesses and key start-ups (roughly 55%) are

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

found to be aware of the fact that the transportation system will be completely autonomous in next 20 years, so they are evolving and transforming their business ideas accordingly.

The rest of the paper is organised as follows: a breif overview of various recent research and development in aid to artificial intelligence and control systems is discussed in Section II. We discuss fundamentals of multi-task learning and meta learning in Section III; a modular end-to-end pipeline of a self-driving car is discussed in Section IV; A practical case-study of comparison between two most popular Self-Driving Cars by Waymo (formerly Google Self-Driving Car Project) and Tesla Motors in Section V; the challenges being faced and possible solutions have been reviewed in Section VI; at last, a brief synopsis of the paper is drawn.

II. Research and Development in AI and control strategies in the field of self-driving cars

Humans started conducting experiments on self-driving cars as early as the 1920s. In 1925, a demonstration of self-driving cars, called ”American Wonder”, was being driven on the streets of New York City, which was being controlled by radio signals. For many years, radio-controlled cars remained a mainstream trend. Then, a Mercedes-Benz robotic van was introduced in Munich, Germany in the 1980s, which was guided by vision. After that, a vision-based guiding system deploying LIDAR, GPS technologies, and computer vision was the focus of research and development. Several methodologies are being developed, and their results are being compared in order to advance research into autonomous systems. The idea of utilising trained neural networks for self-driving cars was introduced in the early 1990s by Dean Pomerleau, a Carnegie Mellon researcher. In this approach, raw images of roads are captured and used for neural network training. Eventually, the steering wheel of the car was controlled according to road conditions. Utilization of neural networks was a breakthrough in the domain of self-driving cars as it was found to be the most efficient method as compared to others. Various training methods for neural networks are available, such as convolutional neural networks (R-CNN). Neural networks usually deploy live video streaming and images of their surroundings for their training. The efficiency of trained neural networks is highly dependent on provided input data and underlaying concepts. The major drawback of the neural network is that its training is only restricted to two-dimensional (2D) images. To overcome the limitations of neural networks, Light Detection And Ranging (LIDAR) technology is being developed, which collects real-time data in 360 degrees of the car’s surroundings. A combination of both LIDAR technology and neural networks is being employed for better efficiency. Moreover, a GPS system was also included for better navigation. As further advancements were made, cars were facilitated with odometry sensors and ultrasonic sensors for motion detection and distance measurement, respectively. A central computer was also embedded in cars, to which all interfaced devices and the car’s controlling systems were connected and controlled by [28].

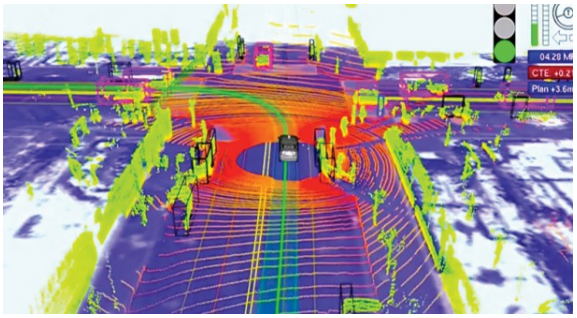

Fig. 1: Application of LIDAR in path planning

The industry of self-driving cars and AD has accomplished many technical advancements in past years. Many researchers and developers are involved in R&D and development activities to bring new innovations and improvements to self-driving cars. Advanced concepts of AI and efficient control strategies are being deployed to achieve the incentives and desired features.

“Autonomous Driving Via Principled Simulations (ADAPS)”, which is a new control policy for self-driving cars, is being proposed in [29]. It is claimed to be robust as it includes various scenarios (also rare scenarios such as accidents) in training data. ADAPS has two simulation platforms, where accidental scenarios are being generated and analysed. As a result, a labelled training dataset as well as a hierarchical control policy are being generated automatically. This memory-enabled and hierarchical control policy can avoid collisions more effectively as it has also been demonstrated by conducted experiments during this research work. Moreover, the online learning mechanism used by ADAPS is declared to be more efficient as compared to other state-of-the-art methods (such as DAGGER) as it requires fewer iterations in the learning process.

Another research work has recommended ”Reinforcement Learning” and ”Image Translation” for self-driving cars [30]. Although reinforcement learning does not require an abundance of labelled data as in supervised learning, one can only train it in a virtual environment. As the real environment includes various unpredictable situations like accidents, Therefore, the real challenge with reinforcement learning is to overcome the gap between the real and the virtual environment. This piece of research has presented an innovative platform for reinforcement learning by deploying an ”Image Semantic Segmentation Network”, which enables the adaptability of the whole model to the real environment. Semantic images tend to discard useless information while focusing on important content required for training and decision-making. Images with semantic segmentation are being used as input to reinforcement learning agents to overcome the difference between real and virtual environments. However, outcomes are found to be restricted by the quality of segmentation technique. In this approach, the model is basically trained in a virtual environment and then transferred to a real environment. Hence, the danger and immense loss associated with experimental failures can be minimized.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

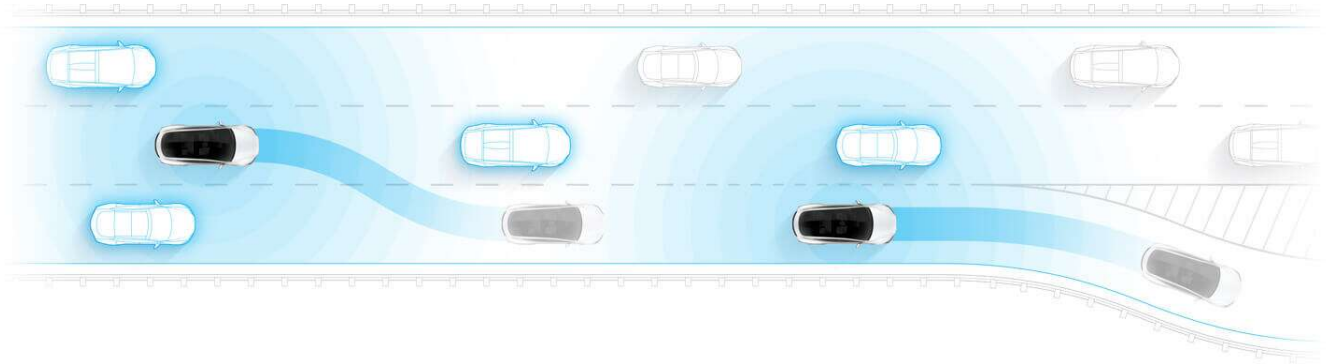

Fig. 2: Predicting Lane Change by Tesla Autopilot

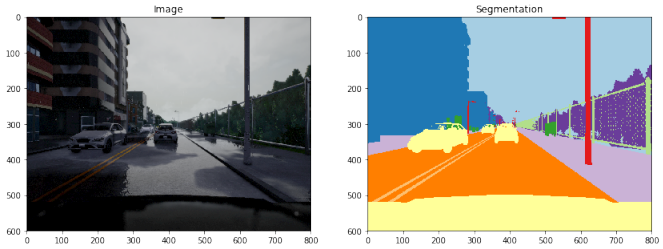

Fig. 3: Semantic Segmentation

The advent of U-Net by Ronneberger et al. [31] was a significant advancement in semantic segmentation architecture. This design is built on the concepts of a conventional Convolutional Network and is separated into two sections: the contracting path and the expanding path. In the contracting route, each convolutional block, together with activation and max pooling, has a stride of 2, resulting in downsampling and doubling the number of feature channels. In the expanding route, the convolutional blocks are upsampled, reducing the number of feature channels into half. These are then joined with their counterparts from the contracting route, followed by 3x3 convolutions and ReLU activation. Finally, a single 1x1 convolution is used to translate the 64-component feature into the required number of classes.

Imitation learning is also a promising technique for self-driving cars. However, one can't simply direct an imitation learning model to accomplish arbitrary goals. For that purpose, goal-directed planning-based algorithms are typically being developed, which deploy reward functions and dynamics models to fulfil arbitrary goals. Yet, the difficulty with this approach is that one can't specify which reward function is evoking the required behaviour. Researchers have tried to combine the upsides of both imitation learning and goal-directed planning and have come up with "Imitative Models" [32]. These are probabilistic and predictive models, which can accomplish specified goals by planning explainable expert-like trajectories. They can learn from the demonstrations of an expert driver without gathering any data online. This proposed "Deep Imitative Model" has been evaluated for various goal objectives such as energy-based goal sets, unconstrained goal sets and constrained goal sets. The model is found to direct its behaviour successfully according to the specified goal. Researchers also claimed that their model outperforms six state-of-the-art imitation learning approaches as well as a planning-based approach in a dynamic self-driving simulation task.Moreover, roughly specified objectives can also be handled by the model with reasonable robustness.

Imitation learning tends to train deep networks according to driving demonstrations of human beings. Such imitation learning models can escape from obstacles and follow straight roads like an expert. But the problem is that an end-to-end driving policy being trained with the help of imitation learning can't be controlled during the testing period and can't be directed to turn at any approaching intersection. An inventive technique called "Conditional Imitation Learning" has been suggested in [33]. High-level commands are being used as input to train this model, which serve two important purposes. During the training period, uncertainties in the perceptuomotor mapping are being resolved by these commands, which promote better learning. While during the testing period, the controller is being directed through the communication channel provided by these commands. This proposed approach is also being evaluated in a dynamic urban simulated environment as well as on a real robotic vehicle (1/5 scale robotic truck). Researchers claimed that performance of command-conditional imitation learning was improved as compared to simple imitation learning under both circumstances.

Researchers have also investigated how the performance of imitation learning can be further improved to achieve robust driving in reality [34]. They claimed that complex scenarios can't be handled by cloning of standard behaviours only (even 30 million examples would not be sufficient). They have suggested synthesising perturbations or interesting situations (for instance, going off the road or collisions) into the expert's driving behavior. In addition, they have proposed to exaggerate actual imitation loss with some extra losses, to discourage failures and unwanted events. These recommendations are found to be ameliorating the robustness of an imitation learning model, named the "ChauffeurNet model". Complex situations can also be handled more efficiently during simulations and ablation experiments.

Recently, the use of various attention mechanisms such as channel, spatial and temporal attention mechanisms have become widely popular due to their inherent similarity with human visual systems. An attention mechanism may be thought of as a dynamic selection process that is implemented by weighting features adaptively based on the relevance of the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

input. Wang et al. [35] introduced self-attention in computer-vision achieving great success in video understanding and object detection. This laid the foundation of numerous works on vision-transformers [36]–[40] which has demonstrated remarkable potential.

Direct perception is another novel approach, recently being proposed for AD. It basically combines the benefits of two other state-of-the-art AD approaches, modular pipelines and imitation learning. In modular pipelines, an extensive model of the environment is being created, while imitation learning does direct mapping of images for output controlling. In direct perception techniques, a neural network is being deployed, which learns from suitable intermediate and low-dimensional representations. The direct perception approaches had already been developed for highway scenarios, but features like following traffic lights, speed limitations, and navigating intersections were still lacking. A recent piece of research work has applied the direct perception approach to the complex urban environment and named it "Conditional Affordance Learning" [41]. It takes video input and maps to intermediate representations, then these high-level directional commands are being used as input for autonomous navigation. Researchers have claimed to achieve 68% betterment in goal-directed navigation as compared to other reinforcement and conditional imitation learning approaches. In addition, other desirable features like smooth car-following, following traffic lights and speed limitations are also being realised in this research work.

A comprehensive and extensive understanding of real traffic is an important prerequisite for self-driving cars. In the near future, various devices such as video recorders, cameras, and laser scanners will be combined to achieve semantic comprehension of traffic. Presently, most of the approaches rely only on large-scale videos for learning purposes, as no proper benchmark for accurate laser-scanner data has been developed yet. An inventive benchmark "Driving Behaviour Net (DBNet)" is recently being presented by researchers [42]. It provides high-quality and large-scale point clouds. A Velodyne laser is being used for scanning and a dashboard camera is deployed for video recording. It also includes the driving behaviour of a standard driver. This extra depth of information provided by the DBNet benchmark dataset has been found to be beneficial for driving policy learning and prediction performance. Similarly, a "LIDAR-Video dataset" has also been presented in [42] to facilitate the detailed understanding of real traffic behaviour.

Another group of researchers have highlighted the significance of "Deep Object-Centric" models for self-driving cars [43]. According to their findings about robotics tasks, when objects are being represented explicitly by a model, then robustness for new scenarios would be higher and visualisations would be more intuitive. Researchers have presented a classification of "Object-Centric" models, which are beneficial for end-to-end learning as well as for object instances. They have also assessed their proposed models in "Grand Theft Auto V simulator" for scenarios where different objects such as pedestrians, vehicles and others are being presented. They have investigated the models according to their selected performance metrics, including collision frequency, interventions, and distance driven. They claimed the outperformance of their models as compared to object-agnostic methods, even when a defective detector is being utilized. Their evaluations in real environments with the "Berkeley DeepDrive Video dataset" also met expectations in low-data regimes.

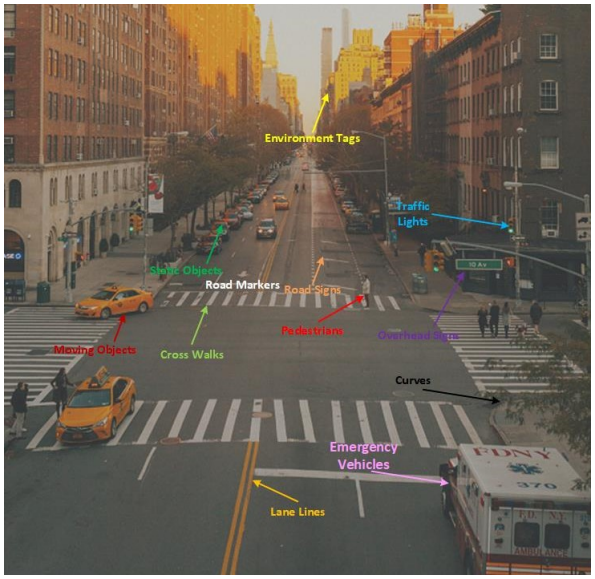

Fig. 4: Various Attributes in Multi-Task Meta Learning Classification

A propitious approach for self-driving cars is end-to-end learning, being investigated by a PHD student [42]. End-to-end learning has the potential to produce desirable self-driving results in complicated urban environments. Representation learning is the main influencer of end-to-end learning protocols, which can be facilitated through auxiliary vision methods (such as semantic segmentation). Naive CNNs can adequately perform reactive control, but complex reasoning can't be accomplished by them. This PHD thesis has mainly focused on handling of scene-conditioned driving, which requires much more than just reactive control. Therefore, he has proposed an algorithmically unified method to combine the benefits of two state-of-the-art methodologies, namely imitation learning and reinforcement learning. His proposed algorithm can learn from its surroundings as well as from demonstration data.

Transferring and scaling end-to-end driving policies from simulation to real-world environments is quite challenging. As the real-world environment has a higher level of complexity as compared to the safe, cheap, and diverse environment being provided by simulations for model training, A novel technique is being discovered by researchers for this purpose, which scales simulated driving policies for real-world environments by deploying abstraction and modularity. The advantages of end-to-end deep learning techniques and modular architecture are being combined by this approach. In this methodology, the core concept is that the driving policy is kept hidden from low-scale vehicle dynamics and unprocessed perceptual input. For evaluation purposes, a driving policy which was being trained on simulations is successfully transferred to a 1/5-scale robotic truck by applying the proposed methodology.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

Moreover, this robotic truck is also being examined on two different continents under various circumstances, and without any finetuning.

The importance of "Surround-View Cameras" and "Route Planners" for end-to-end driving policy learning has also been highlighted in [44]. As humans, we use side-view and rear-view mirrors frequently during driving to get a complete understanding of the vehicle's surroundings. They also rely on their cognitive map while navigating the vehicle over the road. Most of the research and development in the field of self-driving cars deploys only a front-facing camera, without any route planner for driving policy learning. Researchers claim that the information gathered through the mere front-facing camera is not enough for safe navigation. They have presented a better setting which consists of a route planner and a surround-view camera system containing eight cameras. In addition, a CAN bus reader has also been included. This realistic setup of sensors can provide a 360° view of the vehicle's surroundings and an extensive driving dataset including variations in weather conditions and illumination along with miscellaneous driving scenarios. Moreover, a proposed route to the destination and low-scale driving activities (such as predicting speed and steering angle) are also being provided by the sensor setup. For route planning, they deploy "OpenStreetMap" for GSP coordinates and "TomTom Go Mobile" for video recordings of progression. They have empirically demonstrated that the setup with surround-view cameras and route planners has increased the efficiency of AD, especially for urban driving environments with intersections.

Convolutional Neural Networks (CNN) have also been used widely in the domain of self-driving cars. Most of the previously developed end-to-end steering control methods deploy CNN for steering angle prediction using a sequence of images as input. But the whole vehicle's operation can't be controlled by only a single task, learning of steering angles. Therefore, a framework for multi-task learning has been suggested in [45], which simultaneously features speed control and steering angle for end-to-end learning. However, prediction of accurate speeds with only visualisations is not a common practice. This research work has proposed prediction of steering angles and discrete speed commands through a multi-task and multi-modal network, which uses sequences of images, recorded visualisations, and feedback speeds from previous instances as input. For assessment purposes, they have used the "SAIC dataset" and the public Udacity dataset. Their empirical results indicate accurate prediction of speed and steering angle values.

It is a conceivable fact that each possible road scenario can't be modelled explicitly. Basic end-to-end visuomotor policies can successfully be learned through the approach of "Behaviour Cloning", but covering the whole spectrum of behaviours regarding vehicle driving is almost impossible. Important exploratory work has been conducted by researchers [33]. They have developed a benchmark, which can empirically investigate behaviour cloning techniques for their scalability options as well as limitations. Their experimental outcomes indicate that behavior cloning techniques have the potential to perform in arbitrary and unseen situations. Moreover, complicated longitudinal and lateral operations can also be executed without any explicit pre-programming. However, the major limitations associated with behavior cloning protocols are biassing and overfitting of datasets, instability of model training, the presence of dynamic objects, and lack of causality. These shortcomings require further research and technical advancements for practical driving in real-world environments.

Ideally, self-driving policies and models should be examined in real-world environment with the presence of multiple vehicles. But this approach is risky and impractical for enormous amount of research work being conducted in this field. The need for an equitable and proper evaluation method, specifically developed for AD tasks such as "Vision-based Autonomous Steering Control (V-ASC) Models", is also being realized by developers [46]. They have proposed an appropriate dataset for model training purposes. They have also developed a benchmark environment for evaluating different models and generating authentic quantitative and qualitative outcomes. Through this platform, the accuracy of the 'Steering Value' prediction for each individual frame can be examined. In addition, AD results when the frame is changing continuously can also be evaluated. The developers have also introduced a Software-In-the-Loop Simulator (S-ILS)", which can generate an image frame (view-transformed) according to the change in steering values. This generated image depends on the model of the vehicle and the model of the camera sensor. A Baseline model of V-ASC has also been presented, which deploys custom-built features as its base. The proposed simulator and dataset have been evaluated using an end-to-end driving policy based on a convolutional neural network (CNN). The two state-of-the-art methods are being compared, which indicates that the end-to-end CNN technique is more efficient in ground-truth (GT) tracking results, which further depends on human driving outcomes.

The requirement for an appropriate evaluation method is indispensable. Another group of researchers have investigated offline methods of evaluating AD models [33], in which datasets with "Ground Truth Annotation" are collected beforehand for validation purposes. In this piece of research, different offline and online evaluation metrics, specific to AD models, are being related together. According to their findings, offline prediction error doesn't precisely indicate the driving quality of a model. Two models with the same prediction error can demonstrate remarkably different driving performances. However, a suitable selection of validation datasets and appropriate metrics can appreciably enhance the efficiency of offline evaluation methods.

Another innovative approach, "Controllable Imitative Reinforcement Learning (CIRL)", has been proposed for vision-based self-driving cars [47]. Researchers claimed that their proposed driving agent has an improved success rate in high-fidelity simulations, considering it takes only visionary inputs. Classical reinforcement learning methodologies require exploration of continuous and large action space for self-driving tasks, which degrades the efficiency of such approaches. To overcome this problem, action space is being reasonably restricted in this newly proposed technique, CIRL. Encoded

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

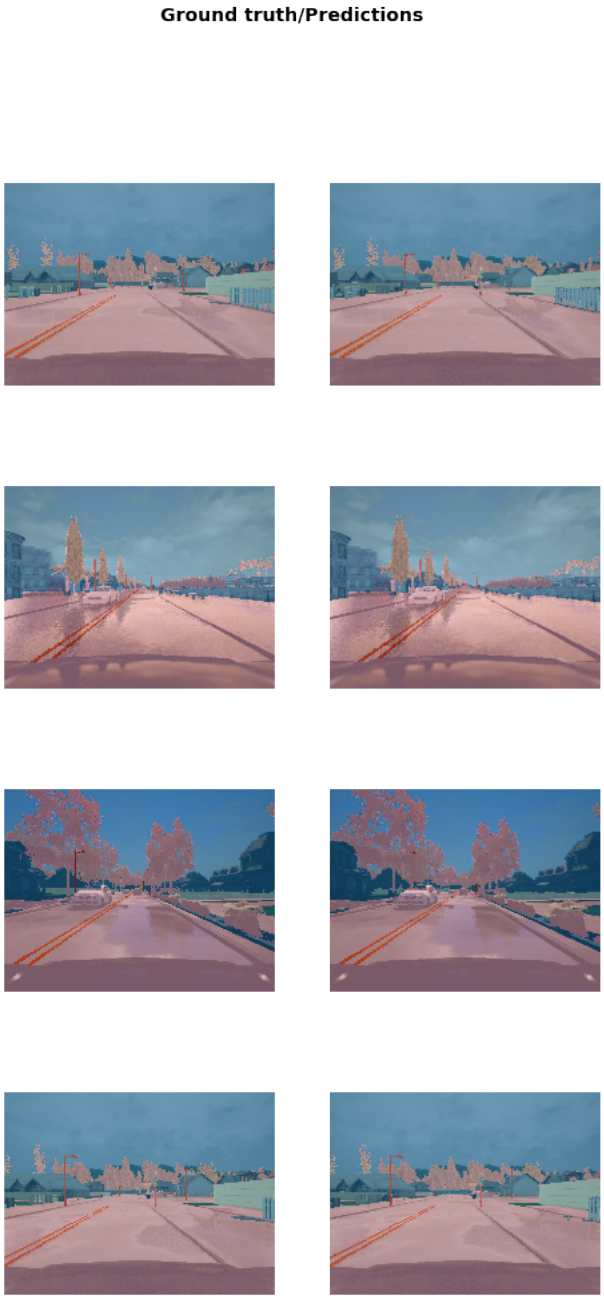

Fig. 5: Semantic Segmentation on CARLA

experiences based on Deep Deterministic Policy Gradient (DDPG) are used to constrain the exploratory action space. In addition, special adaptive strategies and steering-angle reward designs have been proposed for various controlling signals such as ”Turn Right”, ”Turn Left”, ”Follow”, ”Go Straight”. Such proposals depend on the shared representations and enable the model to handle a diversity of scenarios. For evaluation purposes, they deployed ”CARLA Driving Benchmark”. They have experimentally proved that their proposed methodology has better performance in comparison with all previously developed approaches in tackling diverse goal-directed tasks as well as handling unseen scenarios.

Environments with a large number of pedestrians (such as campus surroundings) are known to be troublesome for self-driving policies. Robot or self-driving policies for such an environment need to be intervened in frequently. Robotic decisions which can lead to failure should be attentively tackled in early stages. Therefore, the AD learning policy and the process of data assession for pedestrian-rich environments are quite exacting. A recently conducted research work has proposed the ”Learning from Intervention Dataset Aggregation (DAgger) algorithm” for such surroundings [48]. This algorithm employs an error backtrack function that can be programmed to include expert interventions.The algorithm works in collaboration with CNN and a hierarchically nested procedure for selecting policy. Researchers claim that their proposed algorithm can perform better as compared to standard imitation learning policies in a pedestrian-rich environment. Moreover, there is no need to model the behaviour of pedestrians in the real world explicitly, as control commands can be mapped to pixels directly.

Another promising approach for vision-based self-driving cars in urban areas is “Learning by Cheating” [49]. The developers of this technique have decomposed this complex learning puzzle into two phases. At first stage, some privileged information is being provided to a privileged agent, which also observes locations of all traffic objects and “Ground Truth Layout” of the surroundings. This privileged agent is allowed to cheat with the provided information and its own observations. At second stage, another “Vision-based Sensorimotor” agent gets training from the privileged agent of previous stage. This second stage agent is not allowed to use any privilege information and to cheat. This two-step training algorithm looks illogical at first glance, but its developers have experimentally demonstrated its benefits. They have claimed that their proposed strategy has shown remarkably better performance in comparison with other state-of-the-art methodologies on CARLA benchmark as well as NoCrash benchmark.

Computer Vision, or simply Vision, has been a generally more exploited approach when it comes to the AD environment, and much of it is credited to the plethora of successful research being conducted in Vision. Various types of architecture for object detection models have been published, most of which uses a *Backbone* which is mostly an encoder-decoder and transformer (like VGG16, ResNet50, MobileNet [50], [51], CSPDarkNet53) and a *Head* for actual predictions and Bounding Box (BBox). Multiple architectures have been used for the Head with a Dense Prediction layer only (One-Stage Detector) as well as a Dense and Sparse Prediction layers (Two-Stage Dectector). Examples of One-Stage Detector systems are YOLO (You Only Look Once), SSD [52], RetinaNet [53], CornerNet [54] while Two-Stage Detectors include various R-CNN family, including fast R-CNN, R-FCN [55] and many more. In addition, some research like Feature Pyramid Network (FPN) [56], BiFPN [57], and NAS-FPN [58] have also included an additional *Neck* which basically collects information regarding the feature maps and performs regularization.

Various Data Augmentation techniques like photometric distortion, geometric distortion and object occlusion techniques

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

like CutOut [59], hide-and-seek [60], grid mask [61] have yielded significant improvements in image classification and object detection. Some of these practices involved replacing randomly selected rectangular regions with zeros were applied to feature maps as well like DropOut [62], DropConnect [63] and DropBlock [64]. These techniques were quite successful in dealing with the semantic bias of imbalanced datasets or bias among various classes of labels. Various example mining techniques used in two-stage detection models were able to handle this bias, but could not be applied to the dense prediction layer in one-stage detection systems where focal loss was introduced by Lin [65] to deal with this.

Researchers used Bounding Boxes (BBox) for image classification and object detection where the objective function is mostly a regression task surrounding the object to be detected. Traditional methods used central coordinates, height and width of the BBox, top-left and bottom-right corners or anchor based offset pairs for determining the size and position of the BBox. However, this approach involved treatment of the object as an independent task and the object integrity itself was sometimes compromised. In order to deal with this, researchers introduced IoU loss [66], which is a scale independent method where the ground truth was considered while calculating BBox area. Later, GIoU [67] incorporates the form and orientation of the object, DIoU [68] takes into account the central distance, and CIoU [68] takes into consideration the overlap of the object, the centre points distance, and the aspect ratio, thereby significantly optimizing the BBox accuracy.

Other feature engineering methods like Attention mechanisms, for instance, Squeeze-and-Excitation (SE) [69] and Spatial Attention Module (SAM) [70], and other spatial mechanisms like Spatial Pyramid Pooling (SPP) [71] were proposed, where, information was aggregated from feature maps into a single dimension feature vector. SPP was combined with max-pooling output of kernel size $k \times k, k \in [1, 5, 9, 13]$ and improved YOLOv3-608 AP_{50} by 2.7% on the MS COCO dataset. Further, using RFB [72], it was improved to 5.7%. Furthermore, feature integration techniques like SFAM [73], ASFF [74], and BiFPN [75] were also used, which helped integrate low-level physical features with high-level semantic features on a feature pyramid. Wang proposed PANet [76], which is a few shot non-parametric metric learning system performing segmentation per pixel over the object and learned data. Bochkovskiy et al. [77] proposed YOLOv4, improving the YOLOv3 model by adding SPP block over the CSPDarkNet53 backbone, and using PANet as the path-aggregation neck, Mish-activation [78], DropBlock regularization. In addition, it utilises Mosaic and Self-Adversarial Training (SAT) data aggregation and uses a modified version of SAM, PAN, Cross mini-Batch Normalization (CmBN) which selectively samples data by mini-batches in a given batch and uses CIoU loss in BBox.

III. Multi-Task Learning and Meta Learning

Automatic Machine Learning (AutoML) has developed as a popular subject to make machine learning techniques easier to use and to decrease the requirements of experienced human specialists. Automating laborious activities such as *hyperparameter optimization* (HPO) resulting to greater efficiency, for example *Bayesian optimisation*, might help ML specialists to benefit from AutoML. The renowned AutoML deep learning search from Google is a *Neural Architecture Search* (NAS) [79]. Meta-learning is also a prominent AutoML technology. Meta-learning [80] is the learning science of how ML algorithms learn by leaning a series of methods from earlier models, such as *transfer leaning* [30], *few-shot learning* [81] and even *zero-shot learning* [82].

Loosely speaking, a task $\mathcal{T}$ is a means to achieve an objective by intelligent systems (like robots) by learning from doing the same thing over and over again. While performing a single task learning can be subjected to the environment in which the system acquires the learning goal, it might not apply when the environment is changed. In very simple terms, a supervised learning task (say classification of cat images) using a dataset of animal images $\mathcal{D}$, can be denoted as:

$$\mathcal{D} = \{(\mathbf{x}, \mathbf{y})_k\}$$
$$\min_{\theta} L(\theta, \mathcal{D})$$

where, there are a lot of input-output (x, y) pairs. Here the objective is to minimize any some function function L which is a function of the dataset $\mathcal{D}$ and the parameters of the model θ. The loss function can vary a lot and is often subjected to the learning task. One common choice of such loss function can be a negative log likelihood loss in Equation 1

$$\min_{\theta} L(\theta, \mathcal{D}) = -\mathbb{E}_{(x,y)\sim\mathcal{D}}[\log f_\theta(\mathbf{y}|\mathbf{x})] \tag{1}$$

which is the expectation of the datapoints in the dataset of the log probabilities of the target labels of the model. Thus, a task can be defined as a distribution of the input over the labels and the loss function as in Equation 2.

$$\mathcal{T}_i \triangleq \{p_i(\mathbf{x}), p_i(\mathbf{y}|\mathbf{x}), L_i\} \tag{2}$$

In multi-task learning however, various situations may arise where some of the parameters might be same across all tasks. For example, in multi-task classification, the loss function $\mathcal{L}_i$ is the same across all tasks, and in multi-label learning, both the loss function and the input distribution $(\mathcal{L}_i, p_i(\mathbf{x}))$ is the same across all the tasks. One such example is scene understanding where the loss function can be approximately denoted as in Equation 3. However, the loss functions can vary when the distribution is changed (mixed discrete vs continuous labels) or when one task is preferred more over another task.

$$L_{tot} = w_{\text{depth}} L_{\text{depth}} + w_{\text{kpt}} L_{\text{kpt}} + w_{\text{normals}} L_{\text{normals}} \tag{3}$$

Here, it also uses a task descriptor $\mathbf{z}_i$ which is an index of the task or any meta-data related to the task. Thus, the objective changes to a summation over all the tasks as:

$$\min_{\theta} \sum_{i=1}^{T} L(\theta, \mathcal{D}) \tag{4}$$

also, the some parameters (θ) can be shared and optimized across all the tasks (θ^{sh}) and some parameters can be task-specific parameters (θ^i), which can in turn be shared and optimized among similar sub-tasks. Thus the objective function

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

generalises to Equation 5. Choosing various task descriptor conditioning determines how these parameters are shared.

$$\min_{\theta^{sh},\theta^1,\ldots,\theta^T} \sum_{i=1}^{T} L_i(\{\theta^{sh},\theta^i\},\mathcal{D}_i) \tag{5}$$

A. Conditioning Task Descriptor

Extensive research has been conducted on how to condition the task description with various techniques like:

1) Concatenation: In this technique, the task descriptor is concatenated somewhere in the model (either at the inputs or the activations). Here most of the parameters are shared excluding the metwork components subsequent to the descriptor.

2) Additive Conditioning: Here, the task descriptor is combined with a linear layer and added to the neural network. This is done to match the dimensionality of the task vector to the hidden layers.

3) Multi-head Architecture: Here, the network is split of into different heads of task specific layers. It allows selection of segments of the network to be used for specific tasks.

4) Multiplicative Conditioning: Here, the output of the task descriptor is projected to the individual tasks in the input or the hidden layers. Here each task is trained individually and there is no shared parameters across the tasks. This allows very granular control over features for selective tasks.

While, there are more complex conditioning of task descriptors, each such conditioning technique pertains to the specific optimization problem and requires domain knowledge for efficient utilization.

B. Objective Optimization

Meta Learning follows standard neural network objective function optimization. A simple objective function minimization can be depicted in Algorithm below:

- Sample Mini Batch $\mathcal{B} \sim \{\mathcal{T}_i\}$
- Sample Mini Batch Datapoints for each task $\mathcal{D}_i^b \sim \mathcal{D}_i$
- Compute Loss on Mini Batch $\mathcal{L}(\theta,\mathcal{B}) = \sum_{\mathcal{T}_k \in \mathcal{B}} \mathcal{L}_k(\theta,\mathcal{D}_k^b)$
- Compute Gradient $\nabla_\theta \mathcal{L}$ via Backpropagation
- Optimize using Gradient information

C. One Shot Learning

One-shot learning is a fundamental technique of computer vision, used for categorization of objects present in an image. This approach is in contrast to other machine learning and deep learning techniques, which require abundance of labelled data or thousands of samples to be trained appropriately. One-shot learning, as referred by its name, tries to learn from a single sample or image during training phase and then perform object categorization during testing phase. This technique is inspired from human brain, which can learn thousands of object categories quickly just by observing few samples or examples out of each category. Whenever one-shot learning algorithm is being served with a single example out of a new object category (which is not present in training dataset beforehand), it recognizes this as a new category. Further, it tries to learn out of this single sample and becomes able to re-identify this object class later on. A pertinent application of one-shot learning is face recognition, which is currently being deployed by many smart gadgets and security checks for re-identification purposes. For instance, face recognition system at passport security checks takes passport photo as a single training sample and then identify whether the person owning the passport is the same person as in passport photo [83]. Researchers have developed many algorithms capable of performing one-shot learning. Some of them are:

- 'Probabilistic models' using 'Bayesian learning'
- 'Generative models' deploying 'probability density functions'
- Images transformation
- Memory augmented neural networks
- Meta learning
- Metric learning exploiting 'convolutional neural networks (CNN)' [83]

The field of self-driving cars and robotics is also exploiting one-shot learning algorithms in multiple ways. Generally, deep neural networks are being utilized to perceive driving scenes visually in fully autonomous driving systems. For training purposes, deep neural networks require huge databases, which also need to be annotated manually. A recent research work has proposed one-shot learning, which eliminates the need of manual annotation for training purposes of perception module. A generative framework is being developed called 'Generative One-Shot Learning (GOL)', which receives one-shot samples or generic patterns along with some regularization samples as input. Regularization samples are required to steer the generative process [84]. Similarly, Hadad et al. have recommended one-shot learning techniques for driver identification purposes in shared mobility services. They have used a camera module, python language, and a Raspberry Pi to generate a fully functional system. Face recognition is being performed by one-shot learning approach, which demonstrates satisfactory outcomes [85]. Meanwhile, one-shot learning can also be used to recognize surveillance anomalies. It eliminates the need of precise temporally annotated information. Some researchers have developed an innovative framework to acknowledge surveillance abnormalities by deploying three-dimensional CNN Siamese network. The strong discriminative properties of 3D CNN can highlight the similarities between two sequences of surveillance anomalies [86].

D. Few Shot Learning

Few-shot learning is another sub-domain of machine learning, which learns from a limited number of labelled examples for each classification present in database. Few-short learning mainly comes under supervised learning techniques. The major applicability domain of few-shot learning techniques contains object detection and recognition, image classification as well as sentiment classification. Few-shot learning techniques eliminate the need for gigantic database required by most of the machine learning approaches. By using previously acquired knowledge, few-shot learning has generalizing capabilities for

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

new tasks featuring limited examples with supervised information. However, the main challenge with few-shot learning is unreliability of risk minimizer empirically. Different few-shot learning methods tackle this issue in different ways, which also help categorizing these methods.

- Some methods use data to enhance supervised experience by using previously acquired knowledge
- Some methods use model to minimize the dimensions of hypothesis space by deploying previously acquired knowledge
- Others use previous knowledge to facilitate algorithm which helps in searching of optimal hypothesis present in given space [87]

Few-shot learning techniques have also been adopted by many researchers to achieve certain incentives in the field of autonomous driving and robotics. As with ever-increasing road traffic and diverse traffic scenarios, safety concerns for self-driving cars are becoming more critical. Therefore, a group of researchers have suggested to equip a self-driving car with multiple low-light cameras possessing fish-eye lenses for better assessment of nearby traffic conditions. In addition, front view is being captured by LIDAR and infrared camera. Nearby vehicles and pedestrians in front-view are being identified by few-shot learning algorithm. They have also developed a complete miniature embedded visual assistance system for self-driving cars to accommodate all devices [88]. Similarly, Majee et al. have also encountered 'few-shot object detection (FSOD)' problem for real, class-imbalanced circumstances. They have purposefully selected 'India Driving Dataset (IDD)' as it also contains less frequently occurred road agents. They have experimented FSOD method with both metric-learning and meta-learning and found metric-learning to be superior in outcomes than meta-learning method [89]. Moreover, a novel 'Surrogate-gradient Online Error-triggered Learning (SOEL)' approach has been experimented to achieve few-shot learning. This system operates on neuromorphic processors by combining multiple state-of-the-art techniques such as deep learning, transfer learning as well as computational neuroscience. In this research practice, they trained Spiking Neural Networks (SNNs) partially and made them adaptable online to unseen data classes belonging to a certain domain. SOEL system updates itself automatically after occurring of an error, which allows it to learn fast. Researchers have successfully used their SOEL system for gesture recognition tasks. They claimed that system can online learn new classes of gesture data (other than pre-recorded ones) rapidly with few-shot learning paradigm. Their developed system can also be utilized for behavioural prediction of pedestrians and cyclists in self-driving cars [90]. As enormous point clouds have an ever-increasing applicability in the field of autonomous driving. Many supervised learning tasks have already been accomplished by deep neural networks featuring labelled point clouds. However, point clouds annotation remains an overhead in supervised learning, which should preferably be avoided. Some researchers came up with an innovative idea of using a cover-tree to partition the point cloud hierarchically, which is being encoded by two pre-training jobs of self-supervised learning. They have employed few-shot learning for pre-training of downstream self-supervised learning network [91].

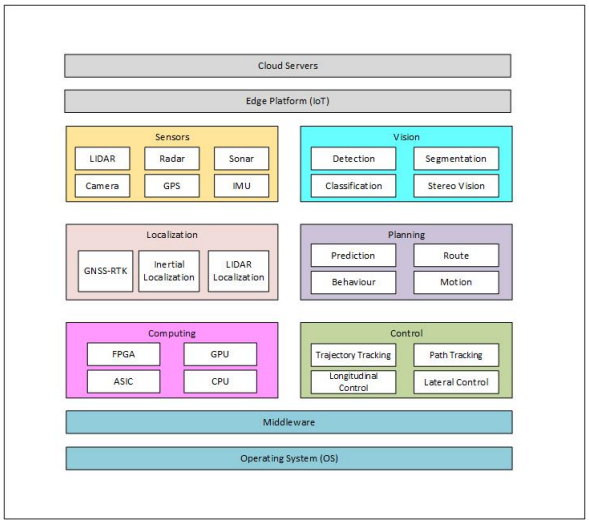

Fig. 6: Self-Driving Car Pipeline

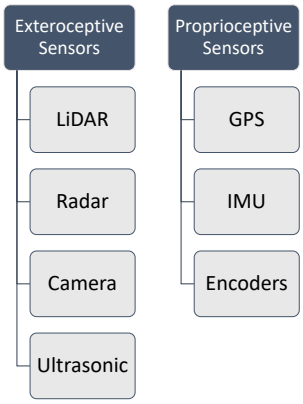

Fig. 7: Sensor Classification

IV. Modular Pipeline

This section presents important modules of self-driving car's pipeline along with their brief descriptions. The picture depicted in Figure 6 shows the pipeline.

A. Sensor Fusion

An important prerequisite of self-driving is to accurately perceive the surrounding area of the vehicle. So that real-time and intelligent driving decisions can be made based on environmental sensing. Various types of sensors are being widely deployed to facilitate self-driving cars with both locational and perceptive surroundings. The two main categories of sensors, along with their varieties, are included in this section.

- Exteroceptive Sensors: They mainly sense the external environment and measure distances to different traffic

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

objects. The following technologies can be used as an exteroceptive sensor in self-driving cars.

  - LiDAR (Light Detection and Ranging): LiDAR can measure distances to different objects remotely by using energy-emitting sensors. It sends a pulse of laser and then senses "Time Of Fight (TOF)", by which pulse comes back.
  - Radar: Radar can sense distances to different objects along with their angle and velocity, by using electromagnetic radiation or radio waves.
  - Camera: A camera builds up a digital image by using passive light sensors. It can detect static as well as dynamic objects in the surroundings.
  - Ultrasonic: An Ultrasonic sensor also calculates distances to neighboring objects by using sound waves.

- Proprioceptive Sensors: They calculate different system values of the vehicle itself, such as the position of the wheels, the angles of the joints, and the speed of the motor.
  - GPS (Global Positioning System): GPS provides geolocation as well as time information all over the world. It is a radio-navigation system based on satellites.
  - IMU (Inertial Measurement Unit): The IMU calculates an object's force, magnetic field, and angular rate.
  - Encoders: It is an electro-mechanical instrument which takes an angular or linear shaft's position as its input and generates a corresponding digital or analogue signal as its output.

As every sensor has its own strength and weakness, such as radar can measure range precisely without being disturbed by environmental illumination but can't deliver enough information about how an object appears or looks like. In contrast, cameras are good at capturing appearance information but can be affected by environmental illumination factors. An intuitive approach to fusing the data provided by different sensors to ameliorate the accuracy and reliability of detection. There are multiple techniques being developed for sensor fusion purposes. Some of them are:

- Vision - LiDAR/Radar: It is used for modelling of surroundings, vehicle localization as well as object detection.
- Vision – LiDAR: It can track dynamic objects by deploying LiDAR technology and a stereo camera.
- GPS-IMU: This system is developed for absolute localization by employing GPS, IMU and DR (Dead Reckoning).
- RSSI-IMU: This algorithm is suitable for indoor localization, featuring RSSI (Received Signal Strength Indicator), WLAN (Wireless Local Area Network) and IMU [92].

B. Localization

Localization methods are required to mention the accurate location of the vehicle on high-definition (HD) maps. HD maps are crucial for self-driving operations as they include significant information about the network of roads, lanes, intersections, signboard positions, etc. in 3D representation. The localization problem requires data from sensors and HD maps as inputs. The following technologies are being used for localization in self-driving cars.

- GNSS-RTk: GNSS (Global Navigation Satellite System) deploys 30 GPS satellites, which are being positioned in space at 20,000 km away from earth. RTK (Real-Time Kinematic) navigation system is also based on satellites and provides accurate position data.
- Inertial navigation: This system is also used for localization and uses motion sensors such as accelerometers, rotational sensors like gyroscopes, and a processing device for computations.
- LIDAR localization: LIDAR sensor provides 3D point clouds, containing information about surroundings. Localization is being performed by incessantly exposing and matching LIDAR data with HD maps. Algorithms used to test point clouds are "Iterative Closet Point (ICP)" and "Filter Algorithms (such as Kalman filter)".

C. Planning and Control

Planning is a decision-taking step and can be considered as a prerequisite for vehicle routing. Routing is basically a driving plan from initial point to destination by using map data, vehicle location on map, and destination as its input. Planning has two major steps:

- Path planning / generation: An appropriate path for vehicle is being planned. If a car needs to change the lane, it must be planned carefully without any accidental scenario.
- Speed planning / generation: It calculates the suitable speed of the vehicle. It also measures the speeds and distances of neighboring cars and utilizes this information in speed planning.

While the control module provides the actual strategy for driving the vehicle on the road, It controls the steering wheel, brakes, and acceleration pedals of the vehicle [1]. Localization and perception modules provide necessary information to planning and control modules for their operation. Principally, one behaviour out of multiple pre-defined behaviours needs to be chosen depending on the available situation. Planning and control modules of self-driving cars typically practise the following protocol.

- The Routing module gets destination data from the user and generates a suitable route accordingly by investigating road networks and maps.
- The behavioral planning module receives route information from the routing module and inspects applicable traffic rules and develops motion specifications.
- The motion planner receives both route information and motion specifications. It also exhibits localization and perception information. By utilising all provided information, it generates trajectories.
- Finally, the control system receives these developed trajectories and plans the car's motion. It also emends all execution errors in the planned movements in a reactive way.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

D. Computer Vision

Computer vision is a special discipline of AI that tends to extract useful information from visionary inputs such as camera outputs, videos, or digital images. This derived information is then processed to make decisions and recommendations accordingly [93]. The concepts of computer vision are also being applied to self-driving cars specifically for localization and perception purposes. "Visual Simultaneous Localization and Mapping (VSLAM)" is an inventive approach to determining a precise vehicle's position in real time. One drawback of the VSLAM technique is cumulative errors, which implies that localization errors increase proportionally with the distance travelled by the vehicle. As a remedy, the VSLAM method is being combined with "Global Navigation Satellite System (GNSS)" localizations for better accuracy. Moreover, active perception is another applicability area of computer vision. "Stereo Vision" can be deployed to derive deep and spatial information about various articles. Similarly, deep learning protocols can provide semantic information about multiple objects. The combination of spatial and semantic data enables the detection of different objects like pedestrians and other vehicles [94].

V. Case Study: Waymo vs Tesla

Extensive research conducted on AD vehicles has made possible actual self-driving vehicles in the real world. These vehicles have been trained for millions of hours in computer simulated environments as well as in real world conditions with actual people inside the vehicles in most cases. Companies like Waymo (formerly Google Self-Driving Car Project) and Tesla Motors have taken it to the upper echelon by rigorously testing their vehicles and deploying a great variety of vehicles, which have travelled for many millions of miles without any serious accidents.

Since the architecture can differ hugely owing to the large combination of underlying framework parameters, pioneers of the trade have used contrasting architectures and deep learning algorithms. Both gather large amounts of meta-data regarding the environment and the surrounding areas, but the processing as well as the data collection methods tend to be different. While organisations like Waymo focus on the standard modular pipeline of an autonomous vehicle (using five cameras on-board, LiDAR and HD maps), other companies like Tesla Motors focus solely on computer vision (using eight cameras only). Doing so increases the amount of computation required on the vision part. However, it simplifies the interaction between different systems in the rest of the pipeline. The meta-data collected can be termed as "trigger-points" which can be used for solving the meta-learning tasks. For instance, such triggers include moving objects, stationary objects, line markings, road markings, crosswalks, road signs, traffic signs, and many other environmental factors as depicted in Figure 4. It is imperative to note that using such vehicles in real world situations would require very high accuracy ($\sim 99.99\%$) in solving these tasks. High reliability in vision requires a very highly accurate model for operation. As such, it can be realised from Figure 8, where a Tesla car can realise

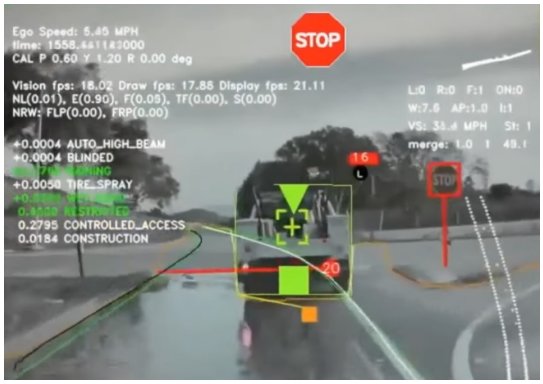

Fig. 8: Real World Tesla Operation

environmental conditions like wet roads in real world and adjust and update its neural network likewise.

Recently, Tesla proposed that they are planning to substitute their previous technology stack consisting of Radar, Vision and Sensor Fusion with Computer Vision alone. The main focus of this kind of approach comes from the fact that Vision is more efficient than their previous Sensor Fusion techniques. asserts that radar has obviously many disadvantages in various conditions, like emergency braking conditions, where the signals represent a jagged response to the sudden brakes. Also, when the vehicle was passing under a bridge, and the radar system was unable to sense whether it was a stationary object or a stationary car, It then purported to Vision for confirmation, which was able to distinguish it by reporting "negative velocity" for a few frames for the stationary object. Furthermore, it also possesses the vertical resolution for classification of a stationary bridge or car. Another major drawback of the Radar stack is that it is more inclined to various environmental triggers. As previously mentioned, radar mostly depends on vision for classification. Hence, a noisy vision layer might cause a delay in response to the radar stack, which in turn reduces the time to reaction to a stationary object. On the contrary, when vision alone is used, it can identify such an object much earlier and result in a longer and smoother braking response. Using vision alone resulted in an increase in precision and recall as compared to the previous stack.

The eight cameras fitted to a Tesla vehicle transmit continuous image sequences. These images are then processed using an image extractor like a standard Resnet model. Following that, all the images from the eight different vision sources are fused together using a Transformer like network. Subsequently, these information is shared across all the cameras and then across all the time. This is achieved using Transformers (like RNN, 3D CNN, or other hybrid structures). Once the multi-camera fusion is processed, the information is passed onto the heads, trunks and finally the terminals. Tesla uses a large amount of branching-like architecture as shown in Figure 10. There are certain advantages to using such a structure as the resources are limited and having an individual neural network for each independent task is not a feasible solution. Thus, feature sharing becomes an important factor in such an architecture as the amount of input data sampled using each input source tends to vary depending on the target

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

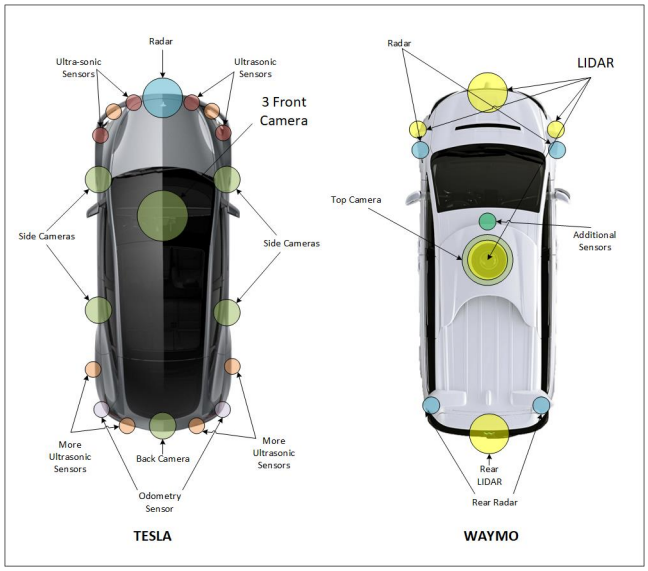

Fig. 9: Various Sensors, Cameras, Radar, LIDAR and other equipments comparison between Tesla and Waymo Vehicles

feature. In addition, it also allows them to decouple the signals at the terminal level independent of operation on other levels.

VI. Challenges

A. Current challenges of self-driving cars

The industry of self-driving cars is currently facing following challenges in the way to attain their potential advantages.

1) Ethical Issues: During the initial phases of the transition from manual driving to self-driving, the existing infrastructure will be shared between self-driving vehicles and human-driven vehicles. There are lots of pedestrians as well, who also need to be considered. Therefore, programmers and designers of self-driving cars must consider a myriad of ethical issues, which increases the design complexity. In particular, when it comes to certain crash scenarios, the ethics followed by self-driving cars have been facing a lot of criticism. Software programmers of self-driving cars need to pay more attention to crash avoidance protocols [95].

2) Cyber Security: Presently, the mainstream research of self-driving cars does not include the issues associated with cybercrimes. However, cybersecurity has great significance for trouble-less functioning and safety in this field, as one hacked vehicle has the potential for huge destruction. There is a dire need to introduce cybersecurity in the operation of self-driving cars themselves as well as communication between them [95].

3) Road Infrastructure and the Transition: The infrastructure of roads is a consequential factor in the transition towards Fully Automatic Driving (FAD). It has even more significance in the initial phases of transition, when self-driving cars and human drivers need to drive on the same roads. Under this scenario, the main challenge is that human reactions tend to be uncertain in response to random events [95]. Self-driving cars would definitely face difficulties while interacting with human drivers. To overcome the problem, construction of autonomous-only traffic lanes has been suggested. These special traffic lanes would be reserved only for self-driving cars until the complete transformation to autonomous driving (AD) is accomplished [96].

4) Regulatory Needs: The government and authorities need to play an important role in the transition towards AD. The government must take such actions which can stimulate the successful realisation of the AD environment [96]. For instance, the government can attract various automotive "Original Equipment Manufacturers (OEMS)" and start-ups through incentives towards AD. Moreover, the government should restrict such regulators' needs which discourage R&D and innovation in the field of AD. Furthermore, legislative partnerships and government agencies can speed up the development in this research domain [2].

5) Hardware Requirements and Resource Allocation: Since autonomous vehicles use various different sources of input data, the amount of data required to process scales up exponentially in a very short amount of time. Thus, it requires the use of specialised on-premise hardware which should be able to solve these arbitrary classification tasks with minute precision. The number of individual classification tasks scales up with the complexity of the system. Allocating individual neural networks to solve each such task can very well lead to a bottleneck. In order to mitigate such situations, companies tend to allow resource sharing among various tasks. In addition, the amount of data sampled from each task can differ largely. This is controlled by varying the batch intervals as some might require less frequency of data sampling while other tasks might require more data to be sampled at higher frequency.

6) Haywire Environment: Most real world situations might be unpredictable and require in-promptu processing. Most AD environments require well-defined markers as input based on which they perform decision making. However, this is seriously challenged in countries where the traffic regulations are not well-defined. For instance, places where multiple vehicles and pedestrians cross the road simultaneously, or where traffic lights and road markers are not properly defined. In such cases, it can be very difficult for a self-driving vehicle to automatically adjust to the situation due to the unpredictability in traffic movement patterns. It is imperative that such conditions be strictly enforced through strict legal traffic guidelines and dedicated road infrastructure services need to be established.

B. User Acceptance and public opinion and how it can be improved further

The full transformation towards an AD environment would only be possible when the public is convinced to accept the change from manual driving to self-driving. As public opinion towards self-driving cars is an important influencer, a myriad of studies and online surveys have been conducted recently to evaluate the user acceptance factor. A crowdsourced online survey has been conducted with 8,862 participants, which demonstrated that 39% of respondents were in favour of AD, while 23% of respondents expressed a negative attitude towards AD [97]. Similarly, another online survey including 5000 participants from 109 countries indicated that 69% of

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 February 2022 doi:10.20944/preprints202202.0123.v1

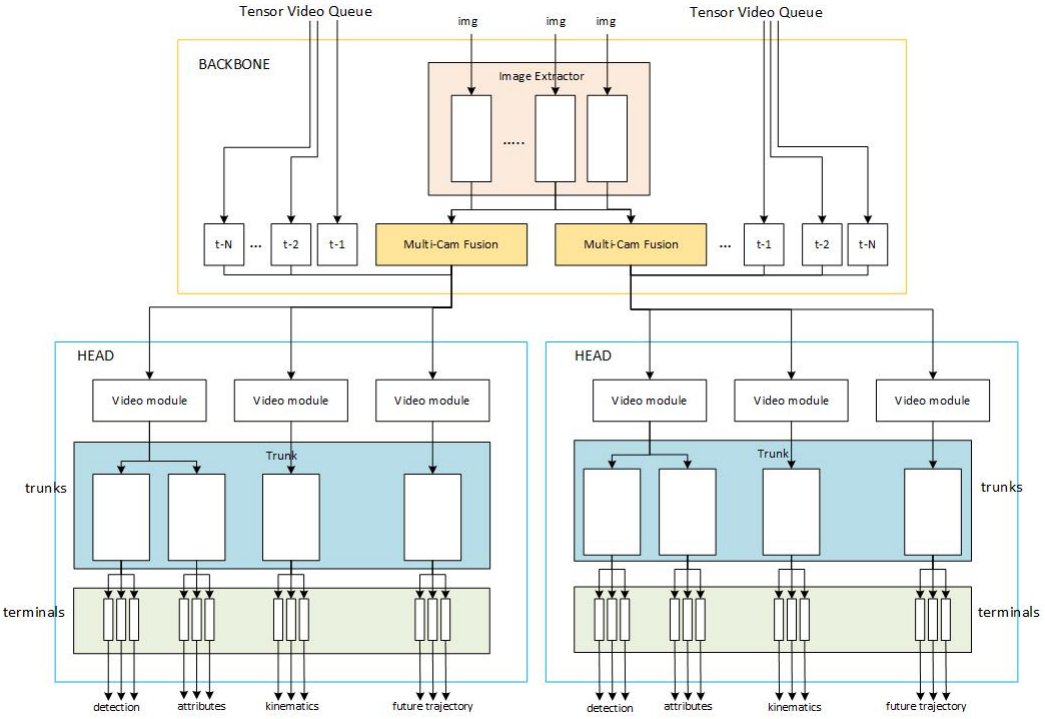

Fig. 10: Tesla Neural Network Architecture

respondents were convinced that the AD industry would have a 50% market share till 2050. However, the main concerns expressed by participants against AD technology were software misuse and hacking, safety, and legal issues [98]. The percentage of user acceptance can be increased by finding the potential solution to the challenges currently faced by industry of self-driving cars. The need for further research, developments, innovations, and technical advancements in this field is indispensable. More reliable and cautious self-driving policies and models would attract more defenders and satisfy the reservations of opposition. As mentioned before in previous section, ethical issues, cyber security, and optimal crash avoidance protocols are mainly the influencers of user acceptance factor.

REFERENCES

[1] A. Yoganandhan, S. Subhash, J. H. Jothi, and V. Mohanavel, "Fundamentals and development of self-driving cars," *Materials today: proceedings*, vol. 33, pp. 3303–3310, 2020.

[2] S. Parida, M. Franz, S. Abanteriba, and S. Mallavarapu, "Autonomous driving cars: future prospects, obstacles, user acceptance and public opinion," in *International Conference on Applied Human Factors and Ergonomics*. Springer, 2018, pp. 318–328.

[3] M. Campbell, M. Egerstedt, J. How, and R. Murray, "Autonomous driving in urban environments: approaches, lessons and challenges," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 368, pp. 4649 – 4672, 2010.

[4] B. Devarajan and R. K. Kapania, "Thermal buckling of curvilinearly stiffened laminated composite plates with cutouts using isogeometric analysis," *Composite Structures*, vol. 238, p. 111881, 2020.

[5] J. Miglani, B. Devarajan, and R. K. Kapania, "Thermal buckling analysis of periodically supported composite beams using isogeometric analysis," in *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2018, p. 1224.

[6] Q. Li, B. Devarajan, X. Zhang, R. Burgos, D. Boroyevich, and P. Raj, "Conceptual design and weight optimization of aircraft power systems with high-peak pulsed power loads," SAE Technical Paper, Tech. Rep., 2016.

[7] B. Devarajan and R. K. Kapania, "Analyzing thermal buckling in curvilinearly stiffened composite plates with arbitrary shaped cutouts using isogeometric level set method," *Aerospace Science and Technology*, p. 107350, 2022.

[8] B. Devarajan, "Thermomechanical and vibration analysis of stiffened unitized structures and threaded fasteners," Ph.D. dissertation, Virginia Tech, 2019.

[9] B. Devarajan, D. Locatelli, R. K. Kapania, and R. J. Meritt, "Thermomechanical analysis and design of threaded fasteners," in *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2016, p. 0579.

[10] J. Miglani, B. Devarajan, and R. K. Kapania, "Isogeometric thermal buckling and sensitivity analysis of periodically supported laminated composite beams," *AIAA Journal*, pp. 1–10, 2021.

[11] S. De and R. K. Kapania, "Algorithms for 2d mesh decomposition in distributed design optimization," *arXiv preprint arXiv:2002.00525*, 2020.

[12] B. Devarajan, "Free vibration analysis of curvilinearly stiffened composite plates with an arbitrarily shaped cutout using isogeometric analysis," *arXiv preprint arXiv:2104.12856*, 2021.

[13] S. De, K. Singh, J. Seo, R. K. Kapania, E. Ostergaard, N. Angelini, and R. Aguero, "Structural design and optimization of commercial vehicles chassis under multiple load cases and constraints," in *AIAA Scitech 2019 Forum*, 2019, p. 0705.

[14] M. Jrad, S. De, and R. K. Kapania, "Global-local aeroelastic optimization of internal structure of transport aircraft wing," in *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2017, p. 4321.

[15] J. H. Robinson, S. Doyle, G. Ogawa, M. Baker, S. De, M. Jrad, and R. K. Kapania, "Aeroelastic optimization of wing structure using curvilinear spars and ribs (sparibs)," in *17th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2016, p. 3994.

[16] S. De, K. Singh, J. Seo, R. K. Kapania, E. Ostergaard, N. Angelini, and R. Aguero, "Lightweight chassis design of hybrid trucks considering multiple road conditions and constraints," *World Electric Vehicle Journal*, vol. 12, no. 1, p. 3, 2021.

[17] S. De, M. Jrad, D. Locatelli, R. K. Kapania, and M. Baker, "Sparibs geometry parameterization for wings with multiple sections using single design space," in *58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2017, p. 0570.

 doi:10.20944/preprints202202.0123.v1

[18] S. De, K. Singh, J. Seo, R. Kapania, R. Aguero, E. Ostergaard, and N. Angelini, "Unconventional truck chassis design with multi-functional cross members," SAE Technical Paper, Tech. Rep., 2019.

[19] S. De, "Structural modeling and optimization of aircraft wings having curvilinear spars and ribs (sparibs)," Ph.D. dissertation, Virginia Tech, 2017.

[20] S. De, K. Singh, B. Alanbay, R. K. Kapania, and R. Aguero, "Structural optimization of truck front-frame under multiple load cases," in *ASME International Mechanical Engineering Congress and Exposition*, vol. 52187. American Society of Mechanical Engineers, 2018, p. V013T05A039.

[21] K. M. Kockelman, P. Avery, P. Bansal, S. D. Boyles, P. Bujanovic, T. Choudhary, L. Clements, G. Domnenko, D. Fagnant, J. Helsel, *et al.*, "Implications of connected and automated vehicles on the safety and operations of roadway networks: a final report," Tech. Rep., 2016.

[22] H. Jamson, N. Merat, O. Carsten, and F. Lai, "Fully-automated driving: The road to future vehicles," in *Proceedings of the Sixth International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design*. Citeseer, 2011, pp. 2–9.

[23] S. De, J. White, T. Brusuelas, C. Patton, A. Koh, and Q. Huang, "Electrochemical behavior of protons and cupric ions in water in salt electrolytes with alkaline metal chloride," *Electrochimica Acta*, vol. 338, p. 135852, 2020.

[24] S. De, W. Sides, T. Brusuelas, and Q. Huang, "Electrodeposition of superconducting rhenium-cobalt alloys from water-in-salt electrolytes," *Journal of Electroanalytical Chemistry*, vol. 860, p. 113889, 2020.

[25] S. De, "Mathematical modeling of cyclic voltammogram curves of copper deposition," 2021.

[26] B. Ponnusamy, "The role of artificial intelligence in future technology," *International Journal of Innovative Research in Advanced Engineering*, vol. 5, no. 4, pp. 146–148, 2018.

[27] P. Stone, R. Brooks, E. Brynjolfsson, R. Calo, O. Etzioni, G. Hager, J. Hirschberg, S. Kalyanakrishnan, E. Kamar, S. Kraus, *et al.*, "Artificial intelligence and life in 2030: the one hundred year study on artificial intelligence," 2016.

[28] V. Shreyas, S. N. Bharadwaj, S. Srinidhi, K. Ankith, and A. Rajendra, "Self-driving cars: An overview of various autonomous driving systems," *Advances in Data and Information Sciences*, pp. 361–371, 2020.

[29] W. Li, D. Wolinski, and M. C. Lin, "Adaps: Autonomous driving via principled simulations," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7625–7631.

[30] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," *ArXiv*, vol. abs/1808.01974, 2018.

[31] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*, 2015.

[32] N. Rhinehart, R. McAllister, and S. Levine, "Deep imitative models for flexible inference, planning, and control," *arXiv preprint arXiv:1810.06544*, 2018.

[33] F. Codevilla, M. Müller, A. López, V. Koltun, and A. Dosovitskiy, "End-to-end driving via conditional imitation learning," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 4693–4700.

[34] M. Bansal, A. Krizhevsky, and A. Ogale, "Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst," *arXiv preprint arXiv:1812.03079*, 2018.

[35] X. Wang, R. B. Girshick, A. K. Gupta, and K. He, "Non-local neural networks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7794–7803, 2018.

[36] M.-H. Guo, J. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S. Hu, "Pct: Point cloud transformer," *Comput. Vis. Media*, vol. 7, pp. 187–199, 2021.

[37] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *ArXiv*, vol. abs/2010.11929, 2021.

[38] L. Yuan, Y. Chen, T. Wang, W. Yu, Y. Shi, F. E. H. Tay, J. Feng, and S. Yan, "Tokens-to-token vit: Training vision transformers from scratch on imagenet," *ArXiv*, vol. abs/2101.11986, 2021.

[39] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," *ArXiv*, vol. abs/2102.12122, 2021.

[40] H. Wu, B. Xiao, N. C. F. Codella, M. Liu, X. Dai, L. Yuan, and L. Zhang, "Cvt: Introducing convolutions to vision transformers," *ArXiv*, vol. abs/2103.15808, 2021.

[41] A. Sauer, N. Savinov, and A. Geiger, "Conditional affordance learning for driving in urban environments," in *Conference on Robot Learning*. PMLR, 2018, pp. 237–252.

[42] Y. Chen, J. Wang, J. Li, C. Lu, Z. Luo, H. Xue, and C. Wang, "Lidar-video driving dataset: Learning driving policies effectively," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5870–5878.

[43] D. Wang, C. Devin, Q.-Z. Cai, F. Yu, and T. Darrell, "Deep object-centric policies for autonomous driving," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8853–8859.

[44] S. Hecker, D. Dai, and L. Van Gool, "End-to-end learning of driving models with surround-view cameras and route planners," in *Proceedings of the european conference on computer vision (eccv)*, 2018, pp. 435–453.

[45] Z. Yang, Y. Zhang, J. Yu, J. Cai, and J. Luo, "End-to-end multi-modal multi-task vehicle control for self-driving cars with visual perceptions," in *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2018, pp. 2289–2294.

[46] S. Kwon, J. Park, H. Jung, J. Jung, M.-K. Choi, I. R. Tayibnapis, J.-H. Lee, W.-J. Won, S.-H. Youn, K.-H. Kim, *et al.*, "Framework for evaluating vision-based autonomous steering control model," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 1310–1316.

[47] X. Liang, T. Wang, L. Yang, and E. Xing, "Cirl: Controllable imitative reinforcement learning for vision-based self-driving," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 584–599.

[48] J. Bi, T. Xiao, Q. Sun, and C. Xu, "Navigation by imitation in a pedestrian-rich environment," *arXiv preprint arXiv:1811.00506*, 2018.

[49] D. Chen, B. Zhou, V. Koltun, and P. Krähenbühl, "Learning by cheating," *ArXiv*, vol. abs/1912.12294, 2019.

[50] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *ArXiv*, vol. abs/1704.04861, 2017.

[51] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520, 2018.

[52] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C.-Y. Fu, and A. Berg, "Ssd: Single shot multibox detector," in *ECCV*, 2016.

[53] T.-Y. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, pp. 318–327, 2020.

[54] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," *ArXiv*, vol. abs/1808.01244, 2018.

[55] B. Hariharan, P. Arbeláez, R. B. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 447–456, 2015.

[56] T.-Y. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 936–944, 2017.

[57] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10 778–10 787, 2020.

[58] G. Ghiasi, T.-Y. Lin, R. Pang, and Q. V. Le, "Nas-fpn: Learning scalable feature pyramid architecture for object detection," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7029–7038, 2019.

[59] T. Devries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *ArXiv*, vol. abs/1708.04552, 2017.

[60] K. K. Singh, H. Yu, A. Sarmasi, G. Pradeep, and Y. J. Lee, "Hide-and-seek: A data augmentation technique for weakly-supervised localization and beyond," *ArXiv*, vol. abs/1811.02545, 2018.

[61] P. Chen, S. Liu, H. Zhao, and J. Jia, "Gridmask data augmentation," *ArXiv*, vol. abs/2001.04086, 2020.

[62] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.

[63] L. Wan, M. D. Zeiler, S. Zhang, Y. LeCun, and R. Fergus, "Regularization of neural networks using dropconnect," in *ICML*, 2013.

[64] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "Dropblock: A regularization method for convolutional networks," in *NeurIPS*, 2018.

[65] T.-Y. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, pp. 318–327, 2020.

[66] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. S. Huang, "Unitbox: An advanced object detection network," *Proceedings of the 24th ACM international conference on Multimedia*, 2016.

[67] S. H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 658–666, 2019.

[68] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-iou loss: Faster and better learning for bounding box regression," *ArXiv*, vol. abs/1911.08287, 2020.

[69] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, pp. 2011–2023, 2020.

[70] S. Woo, J. Park, J.-Y. Lee, and I.-S. Kweon, "Cbam: Convolutional block attention module," in *ECCV*, 2018.

[71] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, pp. 1904–1916, 2015.

[72] S. Liu, D. Huang, and Y. Wang, "Receptive field block net for accurate and fast object detection," *ArXiv*, vol. abs/1711.07767, 2018.

[73] Q. Zhao, T. Sheng, Y. Wang, Z. Tang, Y. Chen, L. Cai, and H. Ling, "M2det: A single-shot object detector based on multi-level feature pyramid network," in *AAAI*, 2019.

[74] S. Liu, D. Huang, and Y. Wang, "Learning spatial fusion for single-shot object detection," *ArXiv*, vol. abs/1911.09516, 2019.

[75] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10 778–10 787, 2020.

[76] K. Wang, J. Liew, Y. Zou, D. Zhou, and J. Feng, "Panet: Few-shot image semantic segmentation with prototype alignment," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9196–9205, 2019.

[77] A. Bochkovskiy, C.-Y. Wang, and H. Liao, "Yolov4: Optimal speed and accuracy of object detection," *ArXiv*, vol. abs/2004.10934, 2020.

[78] D. Misra, "Mish: A self regularized non-monotonic activation function," in *BMVC*, 2020.

[79] T. Elsken, J. H. Metzen, and F. Hutter, "Neural architecture search: A survey," *ArXiv*, vol. abs/1808.05377, 2019.

[80] J. Vanschoren, "Meta-learning: A survey," *ArXiv*, vol. abs/1810.03548, 2018.

[81] Y. Wang, Q. Yao, J. Kwok, and L. Ni, "Generalizing from a few examples: A survey on few-shot learning," *arXiv: Learning*, 2019.

[82] W. Wang, V. Zheng, H. Yu, and C. Miao, "A survey of zero-shot learning," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, pp. 1 – 37, 2019.

[83] N. O'Mahony, S. Campbell, A. Carvalho, L. Krpalkova, G. V. Hernandez, S. Harapanahalli, D. Riordan, and J. Walsh, "One-shot learning for custom identification tasks; a review," *Procedia Manufacturing*, vol. 38, pp. 186–193, 2019.

[84] S. M. Grigorescu, "Generative one-shot learning (gol): A semi-parametric approach to one-shot learning in autonomous vision," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 7127–7134.

[85] M. Haddad, D. Sanders, M. C. Langner, and G. Tewkesbury, "One shot learning approach to identify drivers," in *IntelliSys 2021*. Springer, 2021, pp. 622–629.

[86] A. Ullah, K. Muhammad, K. Haydarov, I. U. Haq, M. Lee, and S. W. Baik, "One-shot learning for surveillance anomaly recognition using siamese 3d cnn," in *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020, pp. 1–8.

[87] Y. Wang, Q. Yao, J. Kwok, and L. Ni, "Generalizing from a few examples: A survey on few-shot learning. arxiv 2019," *arXiv preprint arXiv:1904.05046*, 1904.

[88] S. Liu, Y. Tang, Y. Tian, and H. Su, "Visual driving assistance system based on few-shot learning," *Multimedia Systems*, pp. 1–11, 2021.

[89] A. Majee, K. Agrawal, and A. Subramanian, "Few-shot learning for road object detection," in *AAAI Workshop on Meta-Learning and MetaDL Challenge*. PMLR, 2021, pp. 115–126.

[90] K. Stewart, G. Orchard, S. B. Shrestha, and E. Neftci, "Online few-shot gesture learning on a neuromorphic processor," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 10, no. 4, pp. 512–521, 2020.

[91] C. Sharma and M. Kaul, "Self-supervised few-shot learning on point clouds," *Advances in Neural Information Processing Systems*, vol. 33, pp. 7212–7221, 2020.

[92] S. Campbell, N. O'Mahony, L. Krpalcova, D. Riordan, J. Walsh, A. Murphy, and C. Ryan, "Sensor technology in autonomous vehicles: A review," in *2018 29th Irish Signals and Systems Conference (ISSC)*. IEEE, 2018, pp. 1–4.

[93] J. Janai, F. Güney, A. Behl, A. Geiger, *et al.*, "Computer vision for autonomous vehicles: Problems, datasets and state of the art," *Foundations and Trends® in Computer Graphics and Vision*, vol. 12, no. 1–3, pp. 1–308, 2020.

[94] S. Liu, *Engineering autonomous vehicles and robots: the DragonFly modular-based approach*. John Wiley & Sons, 2020.

[95] S. A. Bagloee, M. Tavana, M. Asadi, and T. Oliver, "Autonomous vehicles: challenges, opportunities, and future implications for transportation policies," *Journal of modern transportation*, vol. 24, no. 4, pp. 284–303, 2016.

[96] J. D. Rupp and A. G. King, "Autonomous driving-a practical roadmap," SAE Technical Paper, Tech. Rep., 2010.

[97] P. Bazilinskyy, M. Kyriakidis, and J. de Winter, "An international crowdsourcing study into people's statements on fully automated driving," *Procedia Manufacturing*, vol. 3, pp. 2534–2542, 2015.

[98] M. Kyriakidis, R. Happee, and J. C. de Winter, "Public opinion on automated driving: Results of an international questionnaire among 5000 respondents," *Transportation research part F: traffic psychology and behaviour*, vol. 32, pp. 127–140, 2015.