Open AutoDriving Whitepaper

RoboDrive.org

Abstract

The autonomous driving industry is undergoing a significant transformation, shifting from traditional rule-based systems to data-driven, end-to-end models. Rule-Based Systems rely on predefined rules and algorithms to navigate driving scenarios. While effective in controlled environments, they often struggle with the unpredictability of real-world situations, requiring extensive management of thousands of corner cases. Meanwhile, end-to-end models leverage AI to learn directly from vast amounts of data. These models interpret sensor data and make driving decisions in real time, providing a flexible and robust solution that adapts to complex driving environments.

Currently, industry leaders exploring end-to-end models, such as Tesla, Waymo, Cruise, Baidu, Huawei, SenseTime, and Xpeng, operate as closed-source proprietary companies. Meanwhile, open-source implementations like OpenPilot face limitations in feature support, resource allocation, and community incentives. The industry is in dire need of a true open-source end-to-end autonomous driving model. To meet this challenge, Auto Driving DAO proposes the OpenAutoDriving (OAD) framework—an end-to-end autonomous driving system designed for flexibility and robustness, and an open-source framework where contributors can collaborate and be fairly rewarded for their contributions.

1 Open AutoDriving Model Technical Details

The Open AutoDriving model is designed to lead the global shift toward data and AI-model driven end-to-end autonomous driving. Our vision is to empower OEMs worldwide with cutting-edge, open-source solutions that accelerate the development and deployment of autonomous driving technologies.

1.1 Overall

The autonomous driving industry is undergoing a significant transformation, shifting from traditional rule-based systems to data-driven, end-to-end models. To understand this shift, the difference between rule-based and end-to-end approaches is:

- Rule-Based Systems: These rely on predefined rules (dealing with thousands of edge cases like what if plastic bag appears on the road) and algorithms to navigate driving scenarios. While effective in controlled environments, they struggle with the unpredictability of real-world conditions.
- End-to-End Systems: In contrast, end-to-end models are data-driven and leverage AI to learn directly from vast amounts of data. These models interpret sensor data and make driving decisions in real-time, offering a more flexible and robust solution capable of adapting to complex and dynamic driving environments.

However, industry leaders exploring end-to-end model like Tesla, Waymo, Cruise, Baidu, Huawei, SenseTime, and Xpeng are closed-source proprietary companies. Open source implementations like openpilot face shortages in feature supports, resource allocation, and incentive for community. The industry needs a true open-source end-to-end auto driving model. To address this major challenge, we propose the Open AutoDriving Model (OAD), an end-to-end autonomous driving system. designed with flexibility and robustness at its core. The architecture integrates multiple input modalities, enabling it to adapt to various sensor configurations.

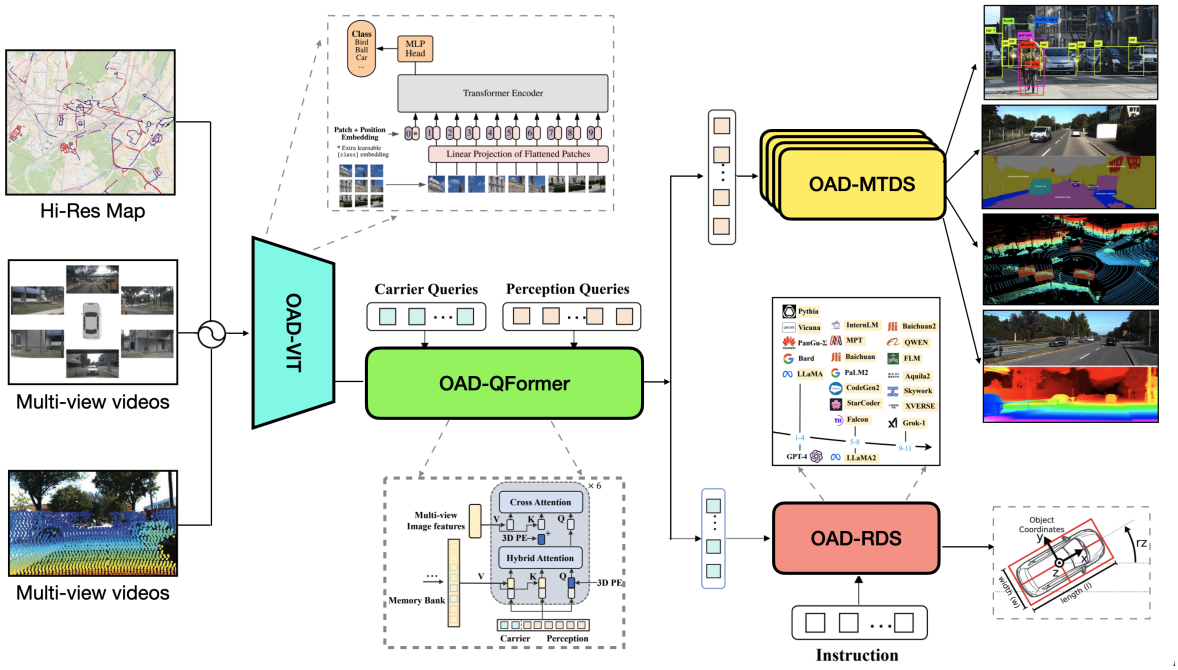

Figure 1: OAD system architecture

1.2 Model Architecture

The proposed autonomous driving system architecture is a end-to-end modular, multimodal framework designed to tackle the complex demands of perception, reasoning, and decision-making in dynamic driving environments. The system is structured into four key components: **the Open AutoDriving Multimodal Perception System (OAD-VIT), the Opeb AutoDriving Visual Language Alignment System (OAD-QFormer), the Open AutoDriving Multi-task Driving System (OAD-MTDS), and the Open AutoDriving Reasoning and Decision System (OAD-RDS).** The OAD-VIT handles sensor fusion and spatial feature extraction through vision, LiDAR, and RiDAR inputs, while OAD-QFormer bridges visual and language modalities to enhance real-time scene understanding. OAD-MTDS enables task-specific predictions for object detection, lane detection, and other critical driving tasks using perception queries. Finally, the OAD-RDS integrates a large language model to facilitate high-level decision-making, path planning, and control generation based on 3D perception. Each module is interconnected through a query-based architecture, allowing efficient information exchange, while maintaining computational efficiency and scalability for diverse driving scenarios.

1.2.1 Open AutoDriving Multimodal Perception System

Motivation Autonomous driving systems require robust multimodal perception to interpret dynamic environments accurately. The motivation behind using a vision transformer [DBK+21] is to effectively handle large-scale, high-resolution, multi-view and multi-type inputs (video, LiDAR, Radar). Traditional perception systems struggle with high-resolution images due to computational constraints and the complexity of 3D environments. Multimodal Vision transformers [WCC+22] are well-suited for extracting spatial and semantic information from images by employing a transformer-based architecture, which allows for handling long-range dependencies and leveraging pre-trained models.

System Design Inspired by [WYJ+24], our proposed Open AutoDriving Vision Transformer (OAD-VIT) is integrated with a multi-camera, LiDAR and RiDAR setup, where each camera provides a different perspective of the driving environment, LiDAR and RiDAR provides precise 3D environment perception. A shared visual encoder extracts features from these multiple views. These features are then transformed into sparse queries through the OAD-QFormer architecture. These queries represent critical 3D spatial features, such as objects and traffic lanes, which are then encoded for further

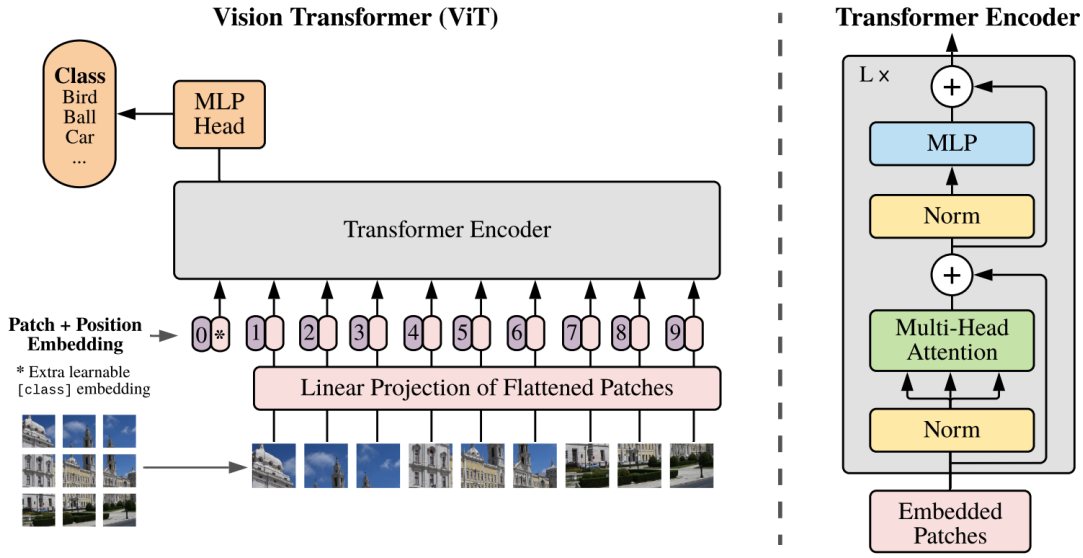

Figure 2: Vanilla Vision Transformer [DBK+21]

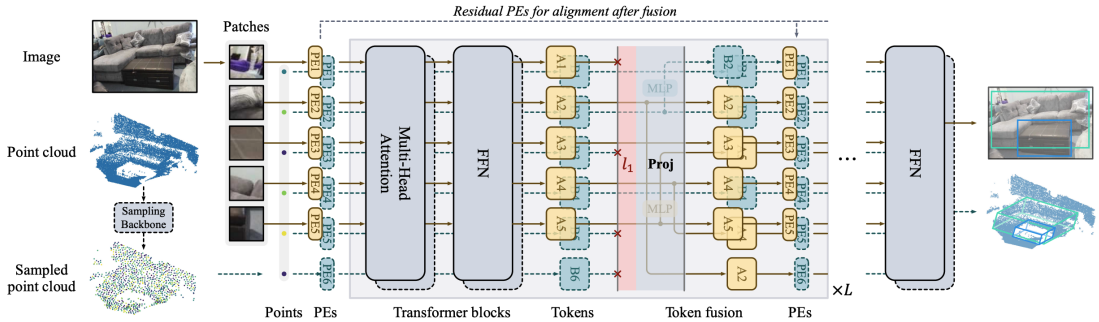

Figure 3: Multimodal Vision Transformer [WCC+22]

processing by other modules in the system. The queries serve as condensed representations of the environment, minimizing computational complexity while maintaining rich semantic information.

Model Detail The implementation begins with multi-view images passed through the visual encoder, generating dense feature maps. These feature maps are encoded with 3D positional information for spatial alignment purposes. A transformer-based Q-Former named OAD-QFormer then compresses the visual data into sparse perception queries, which capture essential information for object detection and scene understanding. These queries are processed to predict object categories, positions, and other spatial attributes. With the progress of visual foundation models [ANK+23], a large number of open source models can be used as visual encoders. In our first implementation, we use EVA-02-L [FSW+24] as the visual encoder. As our modular design concept allows the most advanced visual foundation models to be inserted into OAD-VIT such as [MMM+24], such as CLIP4Clip for video understanding[LJZ+21], ViLD for object detection[GLKC22] , GroupViT for image segmentation[XML+22], DepthCLIP for depth estimation [ZZGL22], and PointCLIP for point cloud processing [ZGZ+21].

The OAD-VIT Multimodal Perception System could be formulated as:

$$F_{Perception} = OAD\text{-}VIT(Vision,\ LiDAR,\ RiDAR) \tag{1}$$

where F is the feature encoded and MVIT is multimodal vision transformer.

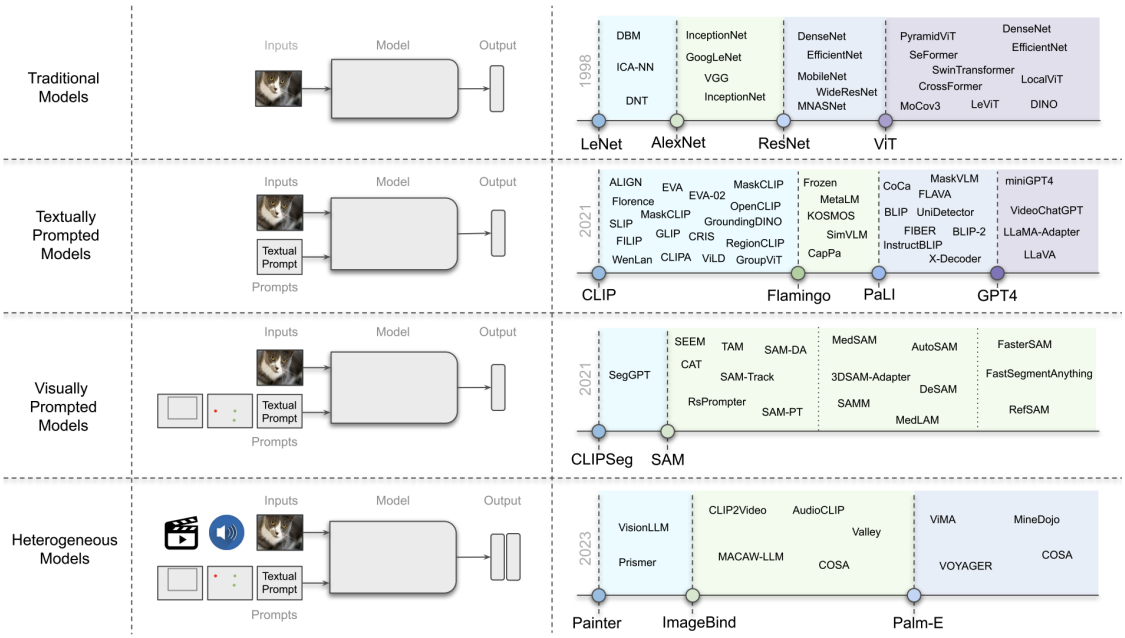

Figure 4: Vision Foundational Models [ANK+23]

1.2.2 AutoDriving Visual Language Alignment System

Motivation A significant challenge in autonomous driving is aligning visual perception with language-based reasoning to enable better scene understanding and decision-making. Although the development of vision-language pre-training models has seen significant advancements in recent years, with larger models achieving state-of-the-art performance across a variety of tasks. However, the high computational costs associated with end-to-end training of such models pose challenges, particularly in real-time applications like autonomous driving, where efficiency and rapid decision-making are critical. To address these challenges, we introduces the AutoDriving-QFormer (OAD-QFormer), an efficient pre-training strategy that leverages frozen image encoders and large pre-trained language models inspired by the BLIP family [LLXH22], [LLSH23], [XSA+24].

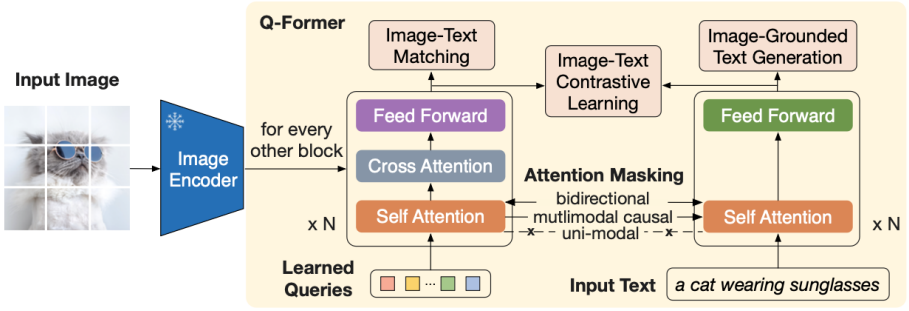

Figure 5: BLIP series Models [LLSH23]

System Design By employing a lightweight Querying Transformer, OAD-QFormer bridges the modality gap between vision and language without requiring extensive computation. This method enables robust vision-language alignment and generation with significantly fewer trainable parameters, making it ideal for resource-constrained environments. In autonomous driving, where timely and accurate interpretation of visual data is essential for tasks such as object recognition, scene under-

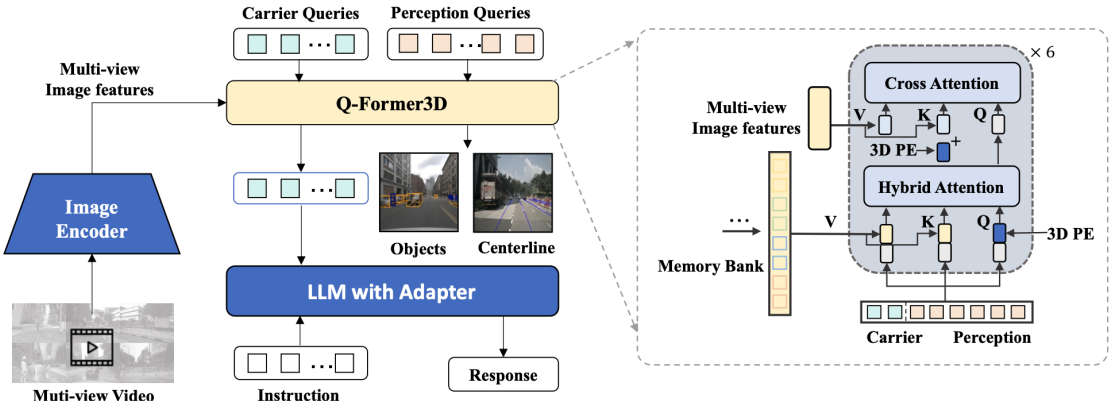

Figure 6: OAD-QFormer Models [WYJ+24]

standing, and decision-making, the compute-efficient architecture of OAD-QFormer offers promising advantages, enhancing the system's ability to process multimodal inputs while maintaining low latency and high accuracy.

Model Details OAD-QFormer is a transformer-based architecture that integrates visual and textual data for reasoning tasks. The system takes multi-view image features and converts them into a sparse set of 3D queries, which are aligned with language tokens. The system starts by extracting features from multi-view images and adding 3D position encoding to the feature maps. A set of carrier queries, which are designed to represent the visual-language alignment, are then passed through a multi-layer transformer decoder. These queries interact with the perception queries by Cross-Attention Mechanism to exchange spatial and semantic information. The resulting carrier queries are aligned with textual inputs, allowing the system to generate language-based reasoning and decision-making outputs based on the 3D perception of the environment.

OAD-QFormer accepts raw multimodal visual data feature, initialized Carrier Queries and Perception Queries; it outputs fine-grained queries that fully understand the visual information through cross-attention mixing. The OAD-QFormer could be formulated as:

$$Q^*_{career}, Q^*_{perception} = OAD - QFormer(F_{Perception}, Q_{career}, Q_{perception}) \tag{2}$$

1.2.3 AutoDriving Multi-task Driving System

Motivation In autonomous driving, the system must simultaneously handle various tasks, such as object detection, lane detection, and motion planning. To address this complexity, we propose the modular AutoDriving Multi-task Driving System: it is a highly scalable module that can include a variety of dedicated models oriented to driving tasks. Each model is fine-tuned at the task level, using perception queries from OAD-QFormer as input and output specific driving information including video understanding[LJZ+21], object detection[GLKC22], image segmentation[XML+22], depth estimation [ZZGL22], point cloud processing [ZGZ+21].

System Design The AutoDriving Multi-task Driving System (OAD-MTDS) utilizes a query-based architecture, where sparse perception queries are initialized to gather information about various traffic elements, such as vehicles, pedestrians, and lane markings. These queries are enhanced with 3D positional encoding and processed through transformer decoder layers to predict the categories and positions of detected objects. By using shared 3D positional encoding, the system efficiently handles multiple tasks simultaneously with minimal computational overhead. For instance, perception queries are initialized for tasks like object detection and lane detection, gathering relevant information from multi-view image features enhanced by 3D positional encoding. The system's shared transformer architecture processes all tasks, with task-specific heads handling the output. One head predicts object positions, while another predicts lane centerlines.

Model Detail The OAD-MTDS is highly scalable and can be expanded according to downstream task requirements. We show four task heads in the figure below, using VILD for Object detection, PointCLIP for point cloud, GroupViT for segmentation, DepthCLIP for depth estimation, etc. More specific tasks are waiting to be added by the open source community. Mathematically, the OAD-MTDS could be described as

$$Task_i = OAD - MTDS - Head_i(Q*_{perception}) \tag{3}$$

where $Head_i$ is a task-specific transformer head.

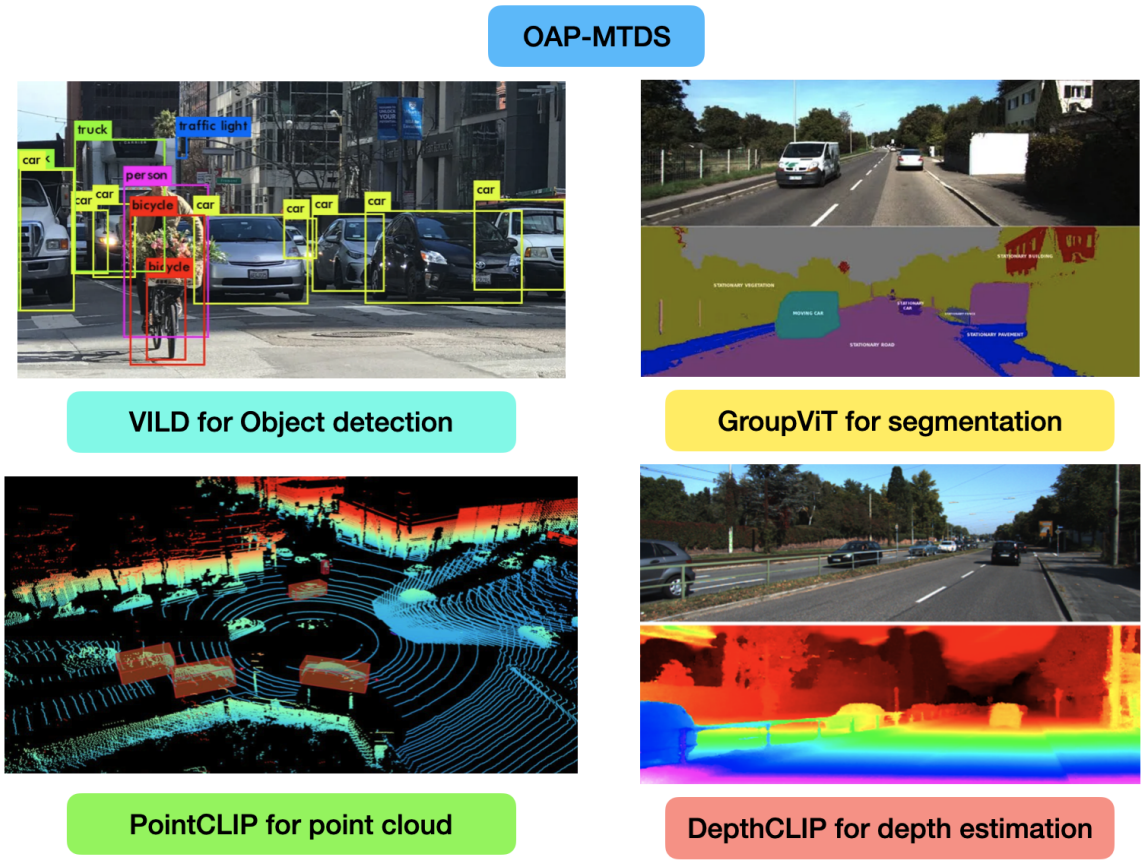

Figure 7: OAD-MTDS

1.2.4 AutoDriving Reasoning and Decision System

Motivation At the heart of autonomous driving tasks are reasoning and decision-making systems that need to have both comprehensive world knowledge and high interpretability. Autonomous driving systems must reason about complex, real-world environments and make decisions based on incomplete or ambiguous information. Recent advances in large language models have made this possible. Large language models (LLMs) offer strong reasoning capabilities, which can be applied to driving scenarios to generate high-level decisions, answer visual questions, and perform counterfactual reasoning. By integrating LLMs with 3D perception, the system can make more informed and context-aware decisions. Hence we propose the Open AutoDriving Reasoning and Decision Making System (OAD-RDS).

System Design The OAD-RDS model integrates a Large Language Model (LLM) for advanced reasoning and decision-making, interpreting 3D scene understanding from the perception system to generate high-level driving decisions. This LLM is pre-trained on extensive datasets and fine-tuned for specific autonomous driving tasks, enabling it to manage complex scenarios with minimal computational overhead. Serving as the reasoning engine, the LLM processes inputs like detected objects,

lane positions, and traffic signals, while reasoning about traffic rules and predicting the best course of action. It is trained to answer questions about the driving scene and generate decision-making outputs based on the current and predicted environment. Most importantly, OAD-RDMS provides a highly explainable human-machine interaction method, allowing humans to fully and in real time understand the current vehicle's perception and decision-making process.

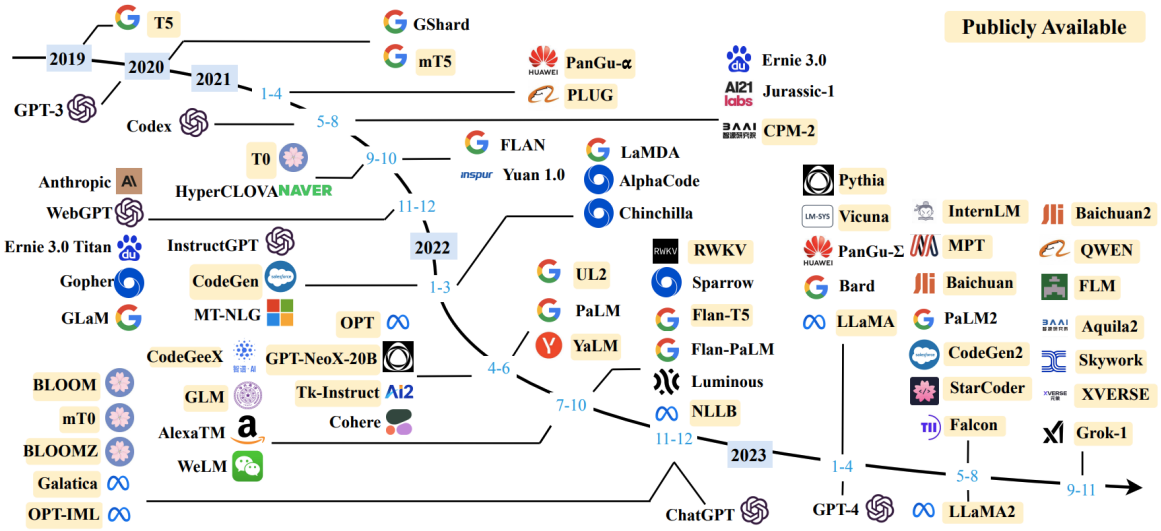

Figure 8: Large Language Model Series [ZZL+23]

Model Detail OAD-RDS takes carrier queries from the OAD-QFormer system, which contain visual and spatial information about the driving scene. These queries are passed through the LLM, which is fine-tuned to handle specific driving-related tasks, such as motion planning, traffic rule validation, and counterfactual reasoning. The system uses pre-trained models for text generation, and additional fine-tuning is performed on driving-specific datasets to ensure the model can handle the nuances of autonomous driving. We can further integrate the capabilities of various open source large language models, using state-of-the-art LLMs such as Llama, Mixtral, Grok series and so on.

$$Description, Reason, Path, Control = OAD - RDS(Q*_{carrier}) \tag{4}$$

Specific driving decisions include the following parts:

Scene Description In this stage, the system first analyzes the surrounding environment by employing multi-modal perception tools. OAD-RDS utilizes a 3D perception system, integrating multi-view image inputs to describe the scene in real-time. The system generates a detailed scene description by recognizing static objects such as traffic signs, road boundaries, and dynamic entities like other vehicles and pedestrians. By extracting and encoding these elements into a structured format, the autonomous agent builds a coherent understanding of the driving environment, which serves as the foundation for further decision-making processes.

Counterfactual Reasoning Counterfactual reasoning is critical for evaluating possible future scenarios by considering alternative actions. In OAD-RDS, this is achieved by simulating different driving trajectories and decisions to anticipate their potential outcomes. For instance, if the system considers changing lanes or accelerating, it uses counterfactual reasoning to evaluate the impact of these actions, such as potential collisions, traffic rule violations, or deviations from the drivable area. This reasoning enables the agent to assess not only the immediate effects of its decisions but also their long-term consequences, enhancing safety and planning accuracy.

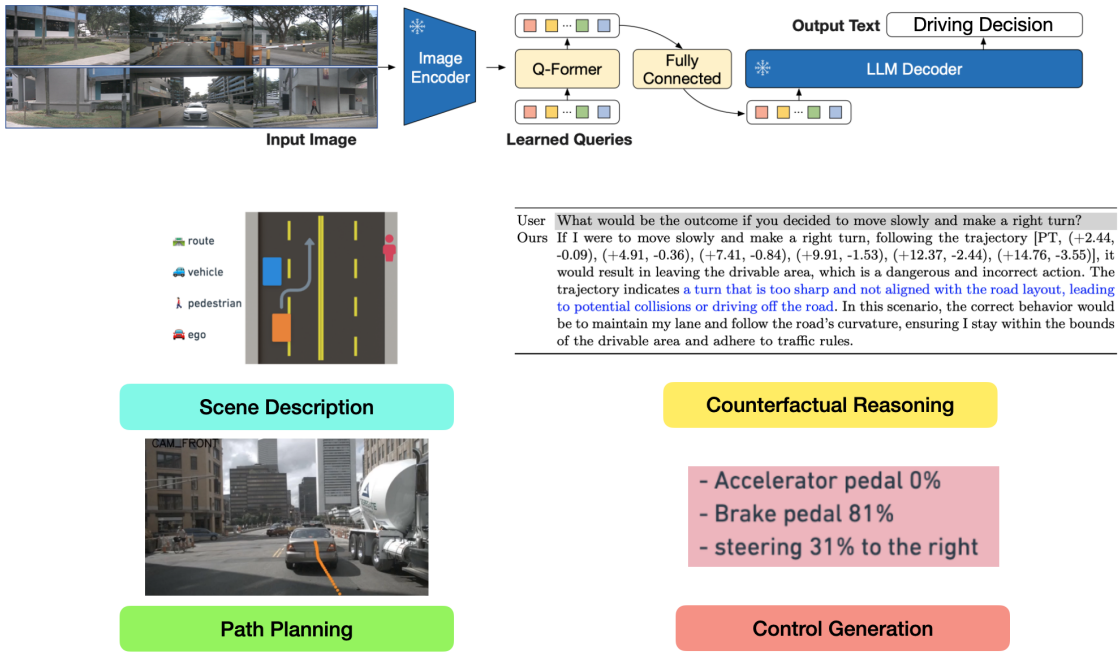

Figure 9: OAD-RDS

Path Planning Path planning involves selecting an optimal route through the environment by considering both the scene description and the outcomes from counterfactual reasoning. OAD-RDS's path planning module calculates possible trajectories while adhering to traffic regulations, avoiding obstacles, and optimizing for safety and efficiency. The system takes into account dynamic entities' movements and static map elements, constructing a plan that ensures smooth navigation through complex urban environments.

Control Generation The final step is control generation, where the system translates the planned trajectory into actionable control signals for the vehicle. OAD-RDS's control generation module continuously adjusts the vehicle's speed, steering, and braking to follow the planned path while responding to real-time environmental changes. This closed-loop control mechanism ensures that the vehicle stays within the drivable area, maintains safe distances from other objects, and reacts promptly to unforeseen hazards.

1.3 Future Development

The future of Auto Driving is centered around pushing the boundaries of autonomous driving through three key advancements: massive data collection, continuous model refinement, and full end-to-end training.

1.3.1 Massive Data Collection: Real World and Simulators

To fully realize the potential of autonomous driving, we will harness vast datasets from both real-world driving environments and high-fidelity simulators. This dual approach accelerates model learning, allowing for rapid iteration and improvement in diverse driving scenarios, including edge cases that are critical for safety and reliability.

1.3.2 Model Fine-Tuning and Upgrading

We envision a dynamic pipeline for continuous model upgrades, integrating feedback from real-time performance to refine the system. This constant fine-tuning, powered by advancements in AI and real-world inputs, ensures that the system stays at the cutting edge of autonomous capabilities.

1.3.3 Fully End-to-End Training

The ultimate goal is a fully end-to-end training regime, where the entire driving stack—from perception to decision-making—learns directly from raw sensor inputs and driving actions. This approach reduces reliance on human design, making the system more adaptive, scalable, and resilient to complex real-world conditions. By embracing this future, we aim to create a self-evolving autonomous driving ecosystem that accelerates the adoption of intelligent, open-source driving solutions worldwide.

References

[ANK+23] Muhammad Awais, Muzammal Naseer, Salman Khan, Rao Muhammad Anwer, Hisham Cholakkal, Mubarak Shah, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Foundational models defining a new era in vision: A survey and outlook, 2023.

[DBK+21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[FSW+24] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *Image and Vision Computing*, 149:105171, September 2024.

[GLKC22] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation, 2022.

[LJZ+21] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval, 2021.

[LLSH23] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, 2023.

[LLXH22] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation, 2022.

[MMM+24] Neelu Madan, Andreas Moegelmose, Rajat Modi, Yogesh S. Rawat, and Thomas B. Moeslund. Foundation models for video understanding: A survey, 2024.

[WCC+22] Yikai Wang, Xinghao Chen, Lele Cao, Wenbing Huang, Fuchun Sun, and Yunhe Wang. Multimodal token fusion for vision transformers, 2022.

[WYJ+24] Shihao Wang, Zhiding Yu, Xiaohui Jiang, Shiyi Lan, Min Shi, Nadine Chang, Jan Kautz, Ying Li, and Jose M. Alvarez. Omnidrive: A holistic llm-agent framework for autonomous driving with 3d perception, reasoning and planning, 2024.

[XML+22] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision, 2022.

[XSA+24] Le Xue, Manli Shu, Anas Awadalla, Jun Wang, An Yan, Senthil Purushwalkam, Honglu Zhou, Viraj Prabhu, Yutong Dai, Michael S Ryoo, Shrikant Kendre, Jieyu Zhang, Can Qin, Shu Zhang, Chia-Chih Chen, Ning Yu, Juntao Tan, Tulika Manoj Awalgaonkar, Shelby Heinecke, Huan Wang, Yejin Choi, Ludwig Schmidt, Zeyuan Chen, Silvio Savarese, Juan Carlos Niebles, Caiming Xiong, and Ran Xu. xgen-mm (blip-3): A family of open large multimodal models, 2024.

[ZGZ+21] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip, 2021.

[ZZGL22] Renrui Zhang, Ziyao Zeng, Ziyu Guo, and Yafeng Li. Can language understand depth?, 2022.

[ZZL+23] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2023.