IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2024.0429000

Transformer-Based Sensor Fusion For Autonomous Vehicles: A Comprehensive Review

AHMED ABDULMAKSOUD[1] AND RYAN AHMED[2]

Centre for Mechatronics and Hybrid Technologies (CMHT), Department of Mechanical Engineering, McMaster University, Hamilton, ON L8S 4L8, Canada

Corresponding author:Ahmed Abdulmaksoud (e-mail: abdula96@mcmaster.ca).

ABSTRACT Sensor fusion is vital for many critical applications, including robotics, autonomous driving, aerospace, and beyond. Integrating data streams from different sensors enables us to overcome the intrinsic limitations of each sensor, providing more reliable measurements and reducing uncertainty. Moreover, deep learning-based sensor fusion unlocked the possibility of multimodal learning, which utilizes different sensor modalities to boost object detection. Yet, adverse weather conditions remain a significant challenge to the reliability of sensor fusion. However, introducing the Transformer deep learning model in sensor fusion presents a promising avenue for advancing its sensing capabilities, potentially overcoming that challenge. Transformer models proved powerful in modeling vision, language, and numerous other domains. However, these models suffer from high latency and heavy computation requirements. This paper aims to provide: (1) an extensive overview of sensor fusion and transformer models; (2) an in-depth survey of the state-of-the-art (SoTA) methods for Transformer-based sensor fusion, focusing on camera-LiDAR and camera-radar methods; and (3) a quantitative analysis of the SoTA methods, uncovering research gaps and stimulating future work.

INDEX TERMS Autonomous driving, artificial intelligence (AI), computer vision, deep learning, machine learning, sensor fusion, transformers

I. INTRODUCTION

AUTONOMOUS vehicles have become one of the major research focuses in modern times due to their potential for reducing costs, improving safety, providing universal access, and offering greater convenience and efficiency compared to conventional vehicles [1]. McKinsey's 2023 global executive survey on autonomous driving reports that investments in autonomy have increased by 30%, reaching more than $11 billion in 2023, to support full-journey autonomous trucks, level-3 highway use cases, and Level 4/5 robo-taxis [2].

Various Autonomous Vehicle (AV) systems exist in the industry, each with slight differences. However, these systems are generally divided into four key elements: perception, localization and mapping, planning, and control [3]. In the perception stage, a group of sensors is used to detect both static and dynamic objects in the environment. Static objects are stationary, while dynamic objects include moving entities such as pedestrians and vehicles. Localization and mapping involve using the data from perception to determine the vehicle's global position relative to world coordinates. Path planning then uses the map generated by localization to calculate the safest and most optimal route for the AV to reach its destination, considering both static and dynamic objects. Finally, the control system follows the waypoints provided by the planning module to manage the vehicle's acceleration, torque, and steering angle. According to the Society of Automotive Engineers (SAE) [4], there are six levels of autonomous vehicles, ranging from level 0, where the driver has full control of the vehicle, to level 5, where the vehicle is fully autonomous as shown in figure 1.

Autonomous perception systems typically integrate multiple sensors such as LiDARs, cameras, and radars to sense the environment. Each of these sensor types has its own strengths and weaknesses as shown in 1, and by fusing them together, the autonomous vehicle can obtain more robust measurements to understand the environment [5], [6]. Table 1 summarizes the comparison among LiDAR, radar, and camera sensors. An example of how sensor measurements can complement each other is as follows:

- Cameras are capable of capturing high-resolution images and details of objects in the scene. However, they struggle in adverse weather conditions that may result in poor lighting, such as fog, heavy rain, or a low-light environment (like nighttime).
- Radar sensors excel at detecting the speed and distance

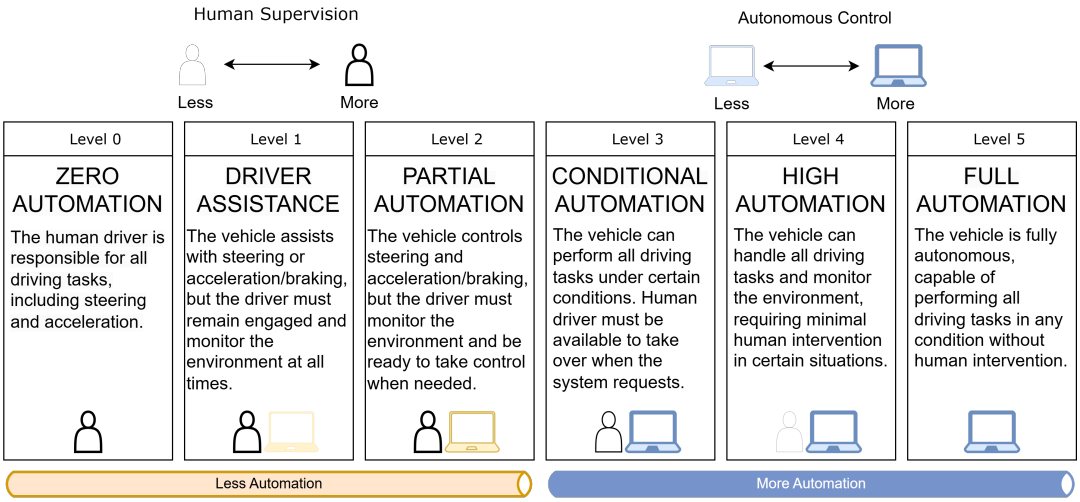

FIGURE 1. **The six levels of driving automation as defined in the Society of Automotive Engineers (SAE) J3016 standard [4].**

TABLE 1. **Comparison between Camera, LiDAR, and Radar**

Criterion	**Camera** e.g., Intel RealSense D455	**LiDAR** e.g., Velodyne 64-line	**Radar** e.g., Continental ARS 408-21
Input	Visual Images	Point Clouds	Radio Waves
Range (m)	≈ 20	≈ 120	≈ 250
Cost	Low	High	Moderate
FPS	≈ 30	≈ 20	≈ 17
Depth Perception	Indirect (via stereo)	High (direct)	Medium (indirect)
Modeling Quality	High (in good lighting)	Medium (sparse compared to camera)	Low (coarse)
Robustness in Weather Conditions	Low	Medium	High
Robustness in Light Conditions	Affected by lighting	Independent of lighting	Independent of lighting
References	[7], [8]	[7], [9]	[7], [10]

of objects in challenging weather conditions, making them more reliable for objects that may not be easily recognized by cameras. However, they produce low-resolution output and cannot provide detailed shapes of the objects in the scene.

- LiDAR sensors offer highly accurate detection of objects and obstacles, along with their distances. They perform moderately across various weather conditions and have a long range with a 360° field-of-view (FoV). However, LiDAR produces a sparse output of point clouds. The higher the LiDAR resolution, the more challenging the processing of the LiDAR points. Additionally, LiDARs may struggle with reflective and transparent surfaces.

As sensor fusion technology has become essential for enhancing the understanding of a vehicle's surroundings, the development of sensor fusion architectures and algorithms has become crucial for ensuring safe autonomous vehicles. Numerous classical algorithms have been developed for data fusion that account for data uncertainty. These methods include probabilistic and fuzzy approaches. Probabilistic methods, such as the Kalman Filter and Particle Filter, along with their variants, use probability to model uncertainty in the information provided. However, they require prior knowledge of the system model and data. Fuzzy methods, such as those discussed in reference [11], are effective in handling uncertainty, imprecision, and non-linear systems, though they rely on extracting knowledge from the data.

Sensor fusion has seen significant advancements in recent years, particularly in the fields of artificial intelligence and machine learning. These advancements have led to the development of new algorithms and applications for multi-sensor data fusion, focusing on improving perception performance and enhancing decision-making processes [12]. Deep learning-based approaches for sensor fusion have been extensively explored across various domains, demonstrating the potential of deep learning algorithms to enhance sensor fusion capabilities. For example, in the context of autonomous vehicle perception and localization, deep learning sensor fusion algorithms have been employed for perception, localization, and mapping, contributing to the safe and efficient operation of autonomous vehicles [13]. Additionally, a deep learning-based radar and camera sensor fusion architecture have been developed for object detection, showcasing the successful integration of deep learning in sensor fusion for advanced applications [14]. Moreover, a deep feature-level sensor fusion framework utilizing skip connections has been proposed for real-time object detection in autonomous driving, highlighting the effectiveness of deep learning in sensor fusion for robust vision-based perception [15]. Furthermore, the application of deep learning in sensor fusion extends to equipment

condition monitoring, where a sensor fusion method using transfer learning models has been developed, emphasizing the utilization of deep learning architectures for effective sensor fusion [16]. Deep learning models have also been investigated for multi-sensor data fusion in bulky waste image classification, exploring early fusion, intermediate fusion, and late fusion techniques within deep learning frameworks [17].

These examples underscore the diverse applications and methodologies of deep learning-based sensor fusion, highlighting the potential of leveraging deep learning algorithms to enhance the integration of multi-sensor data for various purposes. Training deep learning models for sensor fusion presents several challenges, primarily due to the diverse coordinates and modalities of each sensor. The integration of data from multiple sensors with varying modalities requires addressing discrepancies in data representation and feature extraction across different modalities. This necessitates the development of sophisticated fusion techniques capable of reconciling the heterogeneous nature of sensor data [18].

The recent rise of Transformer-based models, first introduced by reference [19], has revolutionized many fields, including language modeling [20], time-series prediction [21], speech recognition [22], and computer vision [23]. Transformers are a type of deep learning architecture that uses attention. Attention allows the model to focus on different parts of the input sequence when producing each part of the output sequence [19]. This approach has proven very useful for tasks where the relationship between the input and output elements is complex and not fixed. Transformer-based sensor fusion revolutionizes autonomous driving by integrating data from multiple sensors like cameras, LiDAR, and radar to enhance perception and decision-making. They excel at feature extraction and representation learning, crucial for handling complex sensor data [24], [25]. For example, fusing RGB images with depth maps improves scene understanding and object detection, leading to safer navigation [26], [27]. Recent work emphasizes combining diverse sensing modalities to leverage their complementary strengths and overcome individual sensor limitations [28], [29]. This multimodal approach boosts robustness and ensures reliable performance in various driving scenarios [13]. Therefore, the application of Transformer models in sensor fusion holds significant promise for enhancing the robustness, accuracy, and efficiency of multi-sensor information integration across various modalities [6]. However, they suffer from high computational demands, potentially increasing the latency of such real-time systems [30].

The remainder of the paper is structured as follows: Section II presents background information on fusion levels and reviews the main transformer architectures—Transformer and Vision Transformer—to ensure smooth reading of the methods discussed in Section III, and reviews the most commonly used datasets for 3D object detection as well as the metrics most frequently used to describe the performance of fusion algorithms. Section III summarizes popular camera-LiDAR and camera-radar transformer-based sensor fusion methods

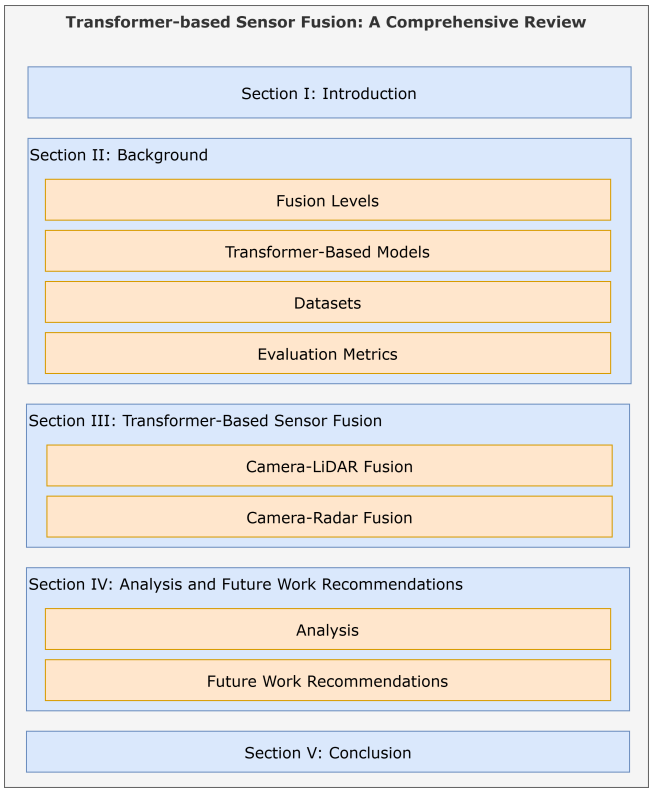

FIGURE 2. An overview of this study

that have been employed in recent years. Section IV provides a comparison between the methods mentioned in Section III and the best-performing methods in each modality, while also identifying research gaps and stimulating future work. Finally, Section V concludes the paper. An overview of the structure of this paper is shown in Figure 2, while Table 2 provides a comparison between this review and other recent, closely related reviews in the literature.

II. BACKGROUND

This section provides background information on fusion levels, transformers, and vision transformers to ensure a smooth reading experience for the methods used that will elaborate on these topics.

A. FUSION-LEVELS

The 3D-detection community classified fusion levels into: detection-level fusion [34], point-level fusion [35], and proposal-level fusion [36]. This classification is based on the incidence of fusion between different modalities (i.e., the timing at which fusion occurs) [6].

Point-level (Early-fusion) augments the point cloud with camera features by establishing hard associations between the points and the images using projection matrices. This approach is inherently semantically lossy as it relies on the quality of the sparse point cloud. Additionally, even minimal noise or errors in the calibration parameters can result in inaccurate fusion [6].

TABLE 2. Comparison between this review and other closely related reviews in the literature

Paper	Year Published	Reviews recent developments in transformers	Reviews work in the literature	Reviews comparison between camera-LiDAR and camera-radar fusion algorithms	Suggests research based on literature gaps	Suggests research based on recent deep learning developments
This paper	2024	✓	✓	✓	✓	✓
[31]	2024		✓		✓	✓
[6]	2023	✓	✓			
[32]	2022		✓		✓	
[33]	2021		✓		✓	

TABLE 3. Comparison of Early Fusion, Deep Fusion, and Late Fusion for Autonomous Vehicles

Aspect	Early Fusion	Deep Fusion	Late Fusion
Definition	Raw sensor data (LiDAR, camera, radar) is combined before processing.	Features from sensors are fused at multiple processing stages.	Decisions from separate sensor models are combined at the end.
Advantages	Captures sensor interdependencies early and is better for low-latency tasks.	Balances early and late fusion benefits while adapting to complex scenarios.	Modular, flexible, and easier to implement.
Disadvantages	High computational cost and complexity with high-dimensional data.	More complex design and requires careful tuning.	Misses early sensor synergies and has limited cross-sensor learning.
Best Use Cases	Real-time sensor fusion for low-latency decisions.	Complex environments needing adaptive fusion.	Tasks with well-defined, independent sensor models.
Examples	Combining raw LiDAR and camera data for object detection.	Adaptive fusion of LiDAR, radar, and camera for dynamic driving scenarios.	Combining separate LiDAR and camera model outputs for final decisions.

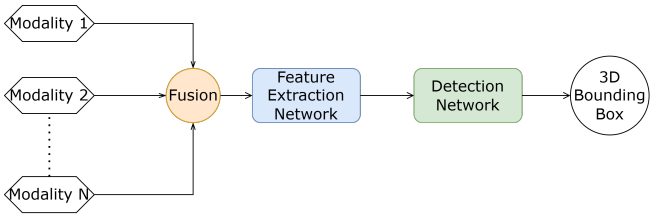

FIGURE 3. Early fusion

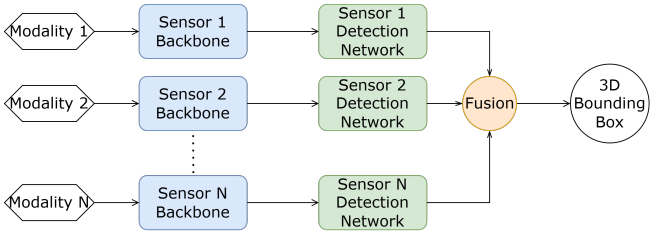

FIGURE 4. Deep fusion

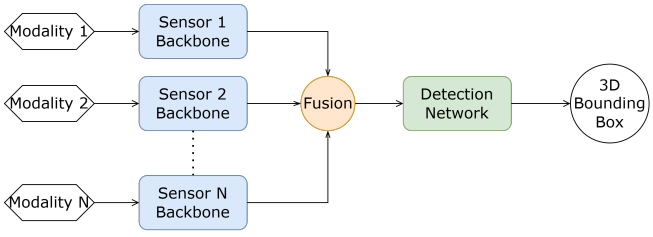

FIGURE 5. Late fusion

FIGURE 6. An overview of the fusion levels [6]

Proposal-level (Deep-fusion) has been extensively researched in the literature [6]. Proposal-level fusion integrates information from different sensors at a higher level of abstraction to achieve accurate detection results. This fusion level aims to exploit multiple related tasks for accurate multi-sensor 3D object detection, combining the results obtained from individual sensors to make a final decision. It allows the fusion of information at a more abstract level, leading to a comprehensive understanding of the environment or object being detected. BEVFusion [37] takes the outputs from image inputs fed into the camera network to generate 3D ego coordinates using Lift-Splat-Shoot (LSS) [38], which are then passed to the BEV encoder to produce camera-based features. Another network processes the 3D point cloud from the LiDAR to generate LiDAR features. Both the LiDAR features and the camera features are fused and fed into the dynamic fusion module, a neural network inspired by the Squeeze-and-Excitation mechanism [39], to produce the fused 3D bounding boxes. In MV3D [40], the LiDAR features are first used to generate initial bounding box estimations. These estimations are then refined using information from the camera features, resulting in more accurate and reliable 3D bounding box predictions. This fusion of LiDAR and camera features enhances the overall performance of object detection and localization in 3D space. TransFuser [41] fuses BEV features from the LiDAR and a single-view image using transformers in the encoder at multiple intermediate feature maps. This enables the output vector to have both global and local representations of the environment. Subsequently, a GRU cell is used to predict differentiable ego-vehicle waypoints, which

are penalized using L1 loss. 4D-Net [42] adds a temporal dimension to the problem and extracts synchronized features from cameras and LiDAR. Then, features are collected at three levels (high-resolution images, low-resolution images, and videos), and cross-modal information is fused using a transformation matrix. The transformation matrix retrieves the 2D context given a 3D center defined by the center point of the grid cell in BEV.

Detection-level (Late-fusion) utilizes Bird's Eye View (BEV) detections of each sensor modality individually after processing. Then, the modalities are combined, and duplicated detections are removed using Hungarian Cost Matching and the Kalman Filter [6]. While this approach leverages multiple modalities for a more robust estimate, it does not exploit the fact that each sensor can also contribute to different interpretations within each prediction, which can provide more information about the detected object. CLOCS [43] work on estimating multimodal fusion by combining output candidates before non-maximum suppression of 2D and 3D detectors, leveraging their geometric and semantic consistencies to produce more accurate final 3D and 2D detections by eliminating false positives.

The fusion levels provide a hierarchical structure for processing sensor data, enabling the integration of information at varying levels of abstraction. Table 3 offers a detailed comparison of these fusion levels. Understanding and effectively implementing these fusion strategies is essential for developing robust and accurate multi-sensor information integration systems across diverse domains. Figure 6 illustrates the key differences between the fusion levels.

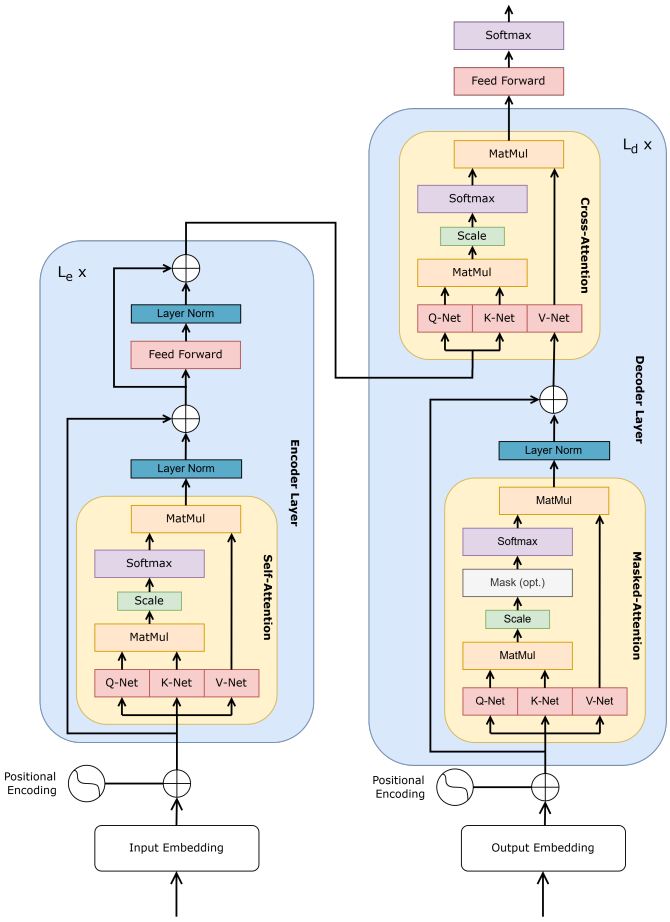

FIGURE 7. Transformer Architecture [19] L_e is the number of encoder layers and L_d is the number of decoder layers

B. TRANSFORMER-BASED MODELS

1) Transformer Model

The Transformer model has three main components: the input embedding with positional encoding, the encoder, and the decoder. First, the input is embedded into a higher-dimensional space across various dimensions. The embedding dimension is represented by d_{model}. Positional encoding is then applied to the embedding to enable the model to utilize the sequence order. This is achieved by applying sine and cosine waves with different frequencies, which make the model context-aware of the signal. Mathematically, the positional encoding is represented by the following:

$$\text{PE}(pos, i) = \begin{cases} \sin\left(\frac{pos}{10000^{2i/d_{\text{model}}}}\right), & \text{if } i \text{ is even,} \\ \cos\left(\frac{pos}{10000^{2i/d_{\text{model}}}}\right), & \text{if } i \text{ is odd.} \end{cases} \quad (1)$$

Here, d_{model} represents the dimension of the model. For an input with dimensions $batch\_size \times N \times L$, where N represents the number of variates and L is the sequence length, the output after applying the embedding and encoding will be $batch\_size \times d_{\text{model}} \times L$.

Multi-head Attention is then applied to the embedded input. Attention, as described in reference [19], is a mapping function that maps a query and a set of key-value pairs into an output. This is accomplished by enabling each attention head to utilize three neural networks to map a portion of the input embedding into *Query* (Q), *Key* (K), and *Value* (V). These values are then used to compute the *Scaled Dot-Product* between Q and K, which is normalized by the dimension (d_k) and passed through a softmax function to determine weights on the output values. The resulting weight matrix is then multiplied by V to obtain the final output. The attention mechanism can be mathematically described as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2)$$

N attention heads are applied to d_{model}/N partitions of the data embedding, allowing each attention head to discern different relationships within the data and enhancing the representation capability of the model. After multi-head attention, a feed-forward neural network with ReLU activation is subsequently applied to the output attention maps. The resulting output of this feed-forward network constitutes the output of a single encoder layer.

The original transformer architecture [19] utilizes a stack of six identical encoder layers, each stacked atop the other.

The architecture of the decoder closely mirrors that of the encoder; however, it incorporates two multi-head attention layers within each decoder layer. The first multi-head attention layer includes a mask to prevent the model from attending to future tokens during the generation process, while the second enables encoder-decoder attention. This arrangement allows the model to focus on pertinent information from the encoder, aiding in the generation of contextually appropriate outputs. The transformer architecture comprises a stack of six decoder layers before applying a fully connected layer to obtain the final output. Figure 7 shows the full architecture of the transformer. Some transformer variants have improved the attention mechanism by reducing its computational complexity to linear, such as the linear attention mechanism introduced in [44].

2) Vision Transformer

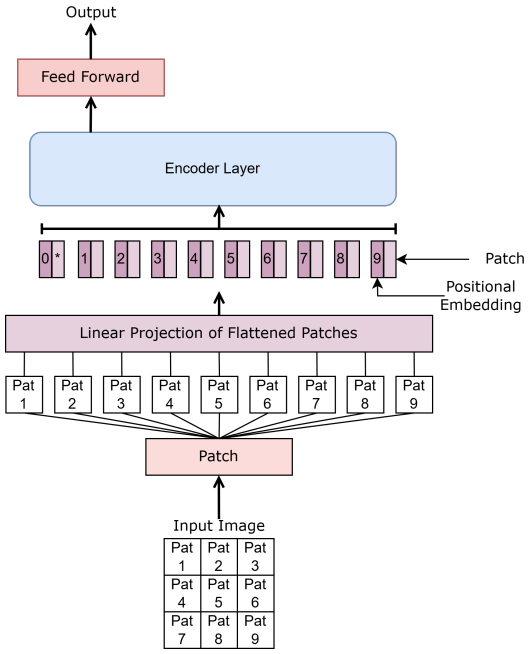

FIGURE 8. Vision Transformer Architecture [45]

The Vision Transformer model [45] shares the same architecture as the original transformer, with a few modifications to process images. The advantage of maintaining most of the original architecture lies in its scalability across various new architectures and their efficient implementations [45]. The Vision Transformer can be divided into four logical components: patching, projection and positional embedding, the encoder layer, and the feedforward layer. Patching is introduced to handle images, as the original transformer was designed for 1D sequential input. This process reshapes 2D images with dimensions $x \in \mathcal{R}^{H\times W\times C}$ into flattened mini-patches $x_p \in \mathcal{R}^{N\times(P^2\cdot C)}$, where H and W are the original image dimensions, C is the number of channels, and the mini-patches are of size $P\times P$. The number of patches is determined by $N = HW/P^2$. Each patch is then flattened to serve as an input token for positional embedding.

The positional embedding is a trainable linear projection that is initially random for each token and then learned for every token. This allows the model to understand the spatial relationships between the input patches. The projection maps the input to a D-dimensional vector, representing the embedding of the images to be provided to the transformer encoder layer. An additional class token is concatenated to the embedded tokens to serve as the image representation, similar to BERT's class token [46]. This representation token learns the structure of the image in the encoder and is later used by the feedforward layer for classification. The classification token, represented by [0,*] in the diagram, is then passed to the feedforward layer to classify the image. Figure 8 illustrates the architecture of the vision transformer with the described operations.

C. DATASETS

Several datasets have been developed to support the research and evaluation of algorithms aimed at enhancing perception capabilities in autonomous vehicles. These datasets encompass a variety of sensor modalities, including LiDAR, cameras, and radar, and are designed to address the challenges associated with real-world driving conditions. For example, recent work [54] introduces a novel domain-adaptive object detection framework that bridges the domain gap caused by varying weather conditions, such as foggy and rainy scenarios, using adaptations at both the image and object levels, along with adversarial techniques. Such advancements underscore the importance of diverse datasets for evaluating algorithms under challenging environmental conditions.

One of the most prominent datasets is the nuScenes dataset [47], which provides a comprehensive multimodal sensor suite consisting of six cameras, five radars, and one LiDAR sensor. This dataset includes 1,000 scenes, each lasting 20 seconds, and is fully annotated with 3D bounding boxes for 23 object classes and eight attributes. The nuScenes dataset is particularly valuable for training and evaluating detection and tracking algorithms, as it captures a wide range of urban driving scenarios [55].

Another significant dataset is the KITTI dataset [48], which has been widely used for benchmarking 3D object detection and localization tasks. The KITTI dataset includes stereo images, LiDAR point clouds, and GPS/IMU data, making it suitable for various sensor fusion applications. It provides annotations for object detection, tracking, and segmentation tasks, thus serving as a critical resource for developing and evaluating sensor fusion techniques [56], [57].

The DENSE dataset [49] is specifically designed to address autonomous driving in adverse weather conditions. Collected in multiple European cities, it includes data from RGB cameras, gated cameras, LiDAR, radar, IMU, and road-friction sensors. This dataset is instrumental in developing robust sensor fusion algorithms that can operate effectively in challenging environmental conditions [58].

TABLE 4. Comparison of Datasets with Adverse Weather, Night, and Day Driving Conditions

Dataset Name	Sensor Modalities	Scenes	Annotations	Adverse Weather	Night Driving	Day Driving
nuScenes [47]	6 Cameras, 5 Radars, 1 LiDAR	1,000	3D Boxes (23 Classes)	✓	✓	✓
KITTI [48]	Stereo Cameras, LiDAR, GPS/IMU	22,000	3D Boxes, Tracking, Segmentation			✓
DENSE [49]	RGB, Gated Cameras, Thermal Cameras, LiDAR, Radar	1,000+	3D Boxes (Multiple Classes)	✓	✓	✓
ApolloScape [50]	Cameras, LiDAR	140,000	3D Boxes, Lane Markings	✓	✓	✓
Cityscapes [51]	Stereo Cameras	5,000	Semantic Segmentation			✓
Waymo Open Dataset [52]	5 LiDARs, 5 Cameras	115,000	3D Boxes (28 Classes)	✓	✓	✓
K-Radar [53]	High-Resolution Radar	1,000+	3D Boxes	✓	✓	✓

The ApolloScape dataset [50] offers a rich set of annotations for various driving scenarios, integrating data from cameras and other sensors. This dataset is particularly useful for training models that require a comprehensive understanding of the driving environment, as it includes diverse weather conditions and times of day [59].

The Cityscapes [51] dataset, while primarily focused on semantic segmentation, provides dense annotations for urban street scenes, which can be beneficial for sensor fusion tasks that require understanding of the environment [60].

The Waymo Open Dataset [52] is another significant resource in the field. It includes high-resolution sensor data collected from autonomous vehicles, featuring a combination of LiDAR and camera data. The Waymo dataset is notable for its large scale and diversity, containing over 115,000 labeled 3D objects across various driving scenarios, making it an essential benchmark for evaluating sensor fusion methods [61].

The K-Radar dataset [53] is specifically designed for radar-based object detection and tracking. It includes high-resolution radar data collected in various driving scenarios, which can be used to develop and evaluate algorithms that leverage radar information for improved object detection and tracking performance. This dataset is particularly valuable for sensor fusion applications where radar data is integrated with other modalities like LiDAR and cameras to enhance detection robustness in diverse environmental conditions [56].

Table 4 highlights the key differences between the datasets. It provides a summary of each dataset's main features, including sensor modalities, number of scenes, annotations, and coverage of diverse conditions such as adverse weather, night driving, and day driving.

D. EVALUATION METRICS

In the context of sensor fusion for 3D object detection, several metrics are commonly employed to evaluate model performance. These metrics are critical for understanding the effectiveness of various sensor modalities, when integrated to improve detection accuracy and robustness.

One of the primary metrics used is Intersection over Union (IoU). It quantifies the overlap between predicted and ground truth bounding boxes. Variants of IoU, such as Generalized IoU (GIoU), have been proposed to address the limitations of the traditional IoU metric, particularly in cases where bounding boxes do not overlap significantly [62]. The IoU is calculated using the following equation:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{3}$$

where the *Area of Overlap* is the area where the predicted and ground truth bounding boxes intersect, and the *Area of Union* is the total area covered by both the predicted and ground truth bounding boxes. The IoU ranges between 0 and 1, with $IoU = 1$ meaning perfect overlap, and $IoU = 0$ meaning no overlap.

Mean Average Precision (mAP) is a standard evaluation metric in object detection tasks. As described in equation 4, it calculates the average precision across different IoU thresholds, providing a comprehensive measure of the model's ability to correctly identify and localize objects in 3D space [63]. This metric is particularly relevant in benchmarking datasets like KITTI [48] and nuScenes [47], where it serves as a key indicator of model performance [64].

$$\text{mAP} = \frac{1}{N}\sum_{i=1}^{N} \text{AP}_i \tag{4}$$

where N is the number of object classes and AP_i is the Average Precision for class i.

Another important metric is the nuScenes Detection Score (NDS), which is a comprehensive metric used to evaluate 3D object detection systems within the nuScenes dataset [47]. NDS combines multiple aspects of detection performance, including mAP and metrics for translation, orientation, rotation, attribute, and velocity errors. This holistic approach captures both object localization accuracy and the quality of object attributes and motion. The NDS is defined as follows:

$$\text{NDS} = \frac{1}{10}[\,5 \cdot \text{mAP} + \sum_{\text{metric}\in\{\text{TP}\}} 1 - \min(1, \text{metric})] \tag{5}$$

where mAP is the mean Average Precision, which measures detection performance, and TP is the set of True Positive metrics (TP metrics), including translation, scale, orientation, velocity, and attribute errors, each normalized between $[0, 1]$.

The TP metrics for evaluating 3D object detection systems measure translation, scale, orientation, velocity, and attribute errors for True Positives (TPs) using a 2-meter center distance threshold during matching. Errors are positive scalar values, averaged across recall levels above 10%. If 10% recall is not achieved, the error is set to 1 for that class [47].

- Average Translation Error (ATE): The Euclidean center distance between the predicted and ground truth objects in meters.

$$\text{ATE} = \|c_{pred} - c_{gt}\| \tag{6}$$

- Average Scale Error (ASE): Defined as $1 - \text{IoU}$ after aligning object centers and orientations.

$$\text{ASE} = 1 - \text{IoU} \tag{7}$$

- Average Orientation Error (AOE): The smallest yaw angle difference between prediction and ground truth, in radians.

$$\text{AOE} = \min(|\theta_{pred} - \theta_{gt}|, 2\pi - |\theta_{pred} - \theta_{gt}|) \tag{8}$$

- Average Velocity Error (AVE): The absolute velocity error in meters per second.

$$\text{AVE} = |v_{pred} - v_{gt}| \tag{9}$$

- Average Attribute Error (AAE): Calculated as $1 - \text{acc}$, where acc is the attribute classification accuracy.

$$\text{AAE} = 1 - \text{acc} \tag{10}$$

Each metric is averaged per class, and the final mean errors are denoted as mATE, mASE, mAOE, mAVE, and mAAE, representing the average errors across all object classes [47].

Root Mean Square Error (RMSE) is also frequently utilized to assess the accuracy of 3D bounding box predictions. RMSE measures the average deviation of predicted values from the actual values, providing insight into the localization accuracy of detected objects [65]. This metric is particularly valuable in sensor fusion contexts, as it can quantify the effectiveness of integrating data from multiple sensors. The RMSE can be calculated as

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{11}$$

where y_i represents the ground truth coordinates, $\hat{y}_i$ represents the predicted coordinates, and n is the number of objects.

Additionally, metrics such as absolute Relative Error (absRel) are employed to evaluate the relative accuracy of depth estimations in 3D object detection tasks. absRel measures the average relative difference between predicted and ground truth values, offering a clear indication of how well the model performs in terms of depth estimation [66]. The absRel can be calculated by:

$$\text{absRel} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{y_i} \tag{12}$$

where y_i is the ground truth coordinate vector, $\hat{y}_i$ is the predicted coordinate vector, and n is the number of objects.

The background concepts introduced in this section, including fusion levels, transformer architectures, and evaluation metrics, provide the foundation for understanding the methods mentioned in Section III. These elements are essential for comparing different approaches and identifying their advantages and limitations. The next section focuses on specific methods, highlighting how these approaches build on the discussed concepts to enhance multimodal perception.

III. TRANSFORMER-BASED FUSION

There are many methods in the literature for transformer-based fusion. Fusion between different sensors may require architectural adjustments. This section reviews some of the architectural differences for sensor fusion across various sensor types. Figure 9 provides an overview of the pipelines used for transformer-based sensor fusion. Each colored path represents a possible way of integrating sensor data after feature extraction. Both the green and red paths illustrate a concatenation of features, where the initial fusion between sensors occurs before entering the transformer decoder. The other paths, orange and blue, show that fusion happens through a cross-attention-based module before proceeding to the prediction network to predict the 3D bounding boxes. Figure 10 shows a timeline figure of the reviewed methods.

A. CAMERA-LIDAR FUSION

TransFusion [67] utilizes a transformer-based architecture to perform LiDAR-Camera fusion for 3D object detection. Unlike traditional methods that rely on hard association based on calibration and projection matrices, TransFusion employs a soft-association mechanism, which is more robust and effective in handling inferior image conditions and sensor misalignment. To detect objects that are challenging to identify in the point cloud, TransFusion uses image-guided query initialization, which helps the LiDAR decoder to better detect these objects. The fusion process is sequential, consisting of two transformer decoder layers. The first layer predicts initial bounding boxes (queries) from the LiDAR BEV features, while the second layer adaptively fuses object queries with useful image features using Spatially Modulated Cross Attention (SMCA). This attention mechanism assigns greater weight to the projected 3D center of each query, enabling the network to focus on pixels close to the object center and ignore irrelevant pixels. This approach allows the network to learn meaningful representations during fusion and reduces training time. Finally, TransFusion uses a transformer-based detection head, where the decoder layer follows a similar design to DETR [80] but with the SMCA, followed by an FFN to generate the final object detections.

Lift [68] on the other hand follows a point-level fusion. They handle the interaction of data from different sensors by performing a 4D sequential data processing scheme. This is done by aligning 4D sequential data for comprehensive multi-modal information aggregation. They also introduce a Sparse Grid-Wise Self-Attention that attends LiDAR point cloud features to the camera features in the BEV space to reduce computational time. This allows for more efficient processing of the data. The features of the Camera and the LiDAR are extracted independently of each other which preserves the modality differences to be processed in later stages. They

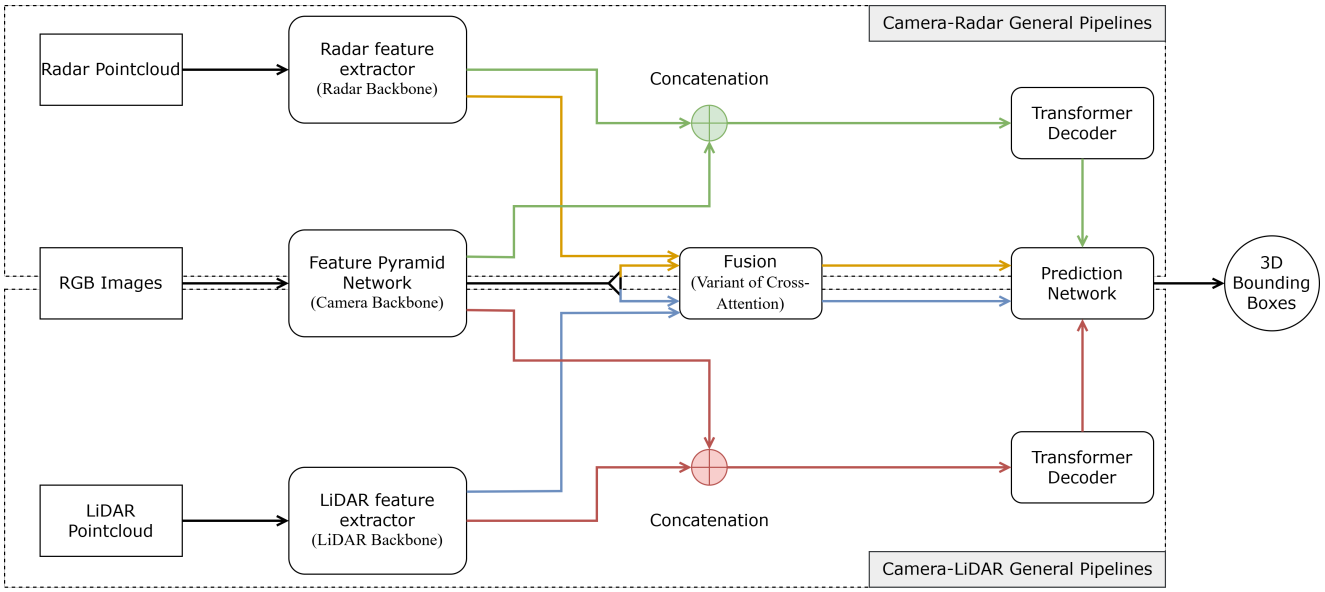

FIGURE 9. **Overview of the transformer-based sensor fusion architecture: The green path shows how camera and radar features are concatenated. The red path demonstrates the concatenation of camera and LiDAR features. The yellow path depicts a cross-attention-based fusion of camera and radar features, while the blue path illustrates a cross-attention-based fusion of camera and LiDAR features.**

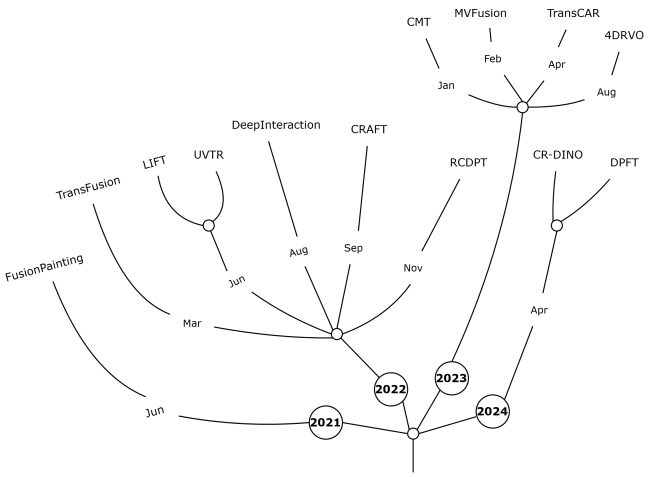

FIGURE 10. **Timeline figure of the reviewed methods.**

perform cross-sensor and cross-time augmentation to ensure consistency across the sensors over time.

Fusion Painting [69] presents a method for combining 2D RGB images and 3D point clouds to detect 3D objects. The method includes three main components: multimodal semantic segmentation at the 2D level, a 3D adaptive attention-based semantic fusion module, and a 3D object detector. Semantic information from 2D images and 3D LiDAR point clouds is obtained using segmentation methods specific to each input. The segmentation results are then combined using the semantic fusion module, enabling the integration of both inputs. This process allows for the labeling of point clouds with fused semantic information in the 3D detector, leading to improved 3D object detection performance compared to using point clouds and images separately.

DeepInteraction [70] proposes a multi-modal 3D object detection framework that maintains separate LiDAR and camera representations throughout the detection pipeline. Unlike other fusion-based approaches, DeepInteraction employs a modality interaction strategy with a multi-modal representational interaction encoder and a multi-modal predictive interaction decoder. This design allows for better exploitation of modality-specific strengths while enabling information exchange between modalities. The method achieves excellent performance on the nuScenes dataset, demonstrating the effectiveness of the interaction-based approach over traditional fusion techniques for multi-modal 3D object detection.

UVTR [71] aims to represent different modalities in a unified framework instead of multi-modality entries. They propose cross-modality interaction by preserving the voxel space without height compression for 3D points. They argue that this method alleviates semantic ambiguity and enables spatial connections, as geometry-aware expressions in point clouds and high contextual features in images are better utilized for improved performance. They use a transformer decoder to efficiently sample features from the unified space with learnable positions, making object-level interactions easier.

Cross-modal Transformer (CMT) [72] integrates information from different modalities without explicitly detailing the fusion process. It employs a transformer model to capture global interactions and relationships implicitly, preserving complementary information across modalities. This is achieved by incorporating cross-modal attention mechanisms that guide the modalities into key-value pairs, facilitating the fusion of cross-modal information. CMT processes images and points cloud tokens as input, directly mapping them to

TABLE 5. Comparison of Multi-Modal Fusion Methods

Method	Fusion Strategy	Modalities	Key Contributions
Camera-LiDAR Fusion			
TransFusion [67]	Transformer-based soft-association, SMCA	LiDAR, Camera	Uses soft-association for robust fusion, image-guided query initialization, and SMCA for efficient feature fusion.
Lift [68]	Point-level fusion, Sparse Grid-Wise Self-Attention	LiDAR, Camera	Aligns 4D sequential data, preserves modality differences, and reduces computational time.
Fusion Painting [69]	Adaptive attention-based semantic fusion	LiDAR, Camera	Combines 2D RGB images and 3D point clouds, uses semantic segmentation for improved detection.
DeepInteraction [70]	Modality interaction strategy	LiDAR, Camera	Maintains separate representations and enables modality-specific strengths.
UVTR [71]	Unified voxel space representation	LiDAR, Camera	Preserves voxel space, uses transformer decoder for efficient feature sampling, and alleviates semantic ambiguity.
Cross-modal Transformer (CMT) [72]	Cross-modal attention mechanisms	LiDAR, Camera	Integrates modalities using a transformer, captures global interactions, and enables fast inference.
Camera-Radar Fusion			
RCDPT [73]	Early and late fusion strategies	Radar, Camera	Combines radar and camera representations, uses vision transformer for depth estimation.
TransCAR [74]	Two-module approach	Radar, Camera	Extracts 2D features, learns radar features, and uses transformer decoder for interaction.
4DRVO [75]	Cross-modal transformer	4D Radar, Camera	Combines sparse radar point clouds with camera images, uses feature pyramid network.
CRAFT [76]	Early fusion with polar coordinates	Radar, Camera	Associates image proposals with radar points, uses cross-attention for information exchange.
MVFusion [77]	Semantic Aligned Radar Encoder	Radar, Camera	Aligns radar features with visual semantics, uses Radar-Guided Fusion Transformer.
CR-DINO [78]	Swin-transformer-based architecture	Radar, Camera	Projects radar points into camera images, uses Swin-transformer for fusion.
DPFT [79]	Radar cube projection	Radar, Camera	Leverages lower-level radar data, incorporates elevation information for improved accuracy.

3D bounding boxes. This approach achieves a 74.1% NDS with fast inference time.

B. CAMERA-RADAR FUSION

RCDPT [73] presents an elegant method for fusing radar and camera data for depth estimation. They propose a fusion strategy that transforms radar data into a dense prediction transformer network by combining camera and radar representations, with radar contributing additional depth information to the monocular camera. They use a vision transformer at the core of their architecture and demonstrate two approaches for fusion: early fusion, where inputs are concatenated before being fed into the embedding, and late fusion, where each input has its own transformer for representation learning before being fused together at the end.

TransCAR [74] proposes a method for 3D object detection using a two-module approach. The first module extracts 2D features from camera images and uses a sparse set of 3D object queries to index the features. These queries then interact with each other using self-attention. The second module learns radar features from multiple scans and uses a transformer decoder to learn the interactions between radar features and vision-updated queries. This allows the model to learn the connections between the two sensors. The method utilizes Hungarian matching loss to penalize the model during optimization.

4DRVO [75] uses a 4D radar to perform visual odometry by combining sparse point cloud data from the 4D radar with image information from the camera sensor. Similar to many other techniques, they use a feature pyramid network alongside pose warping and cost volume to extract robust image and radar features, then feed them into a cross-modal transformer to fuse the two features at multiple scales. They also introduce

a point-confidence estimation module to mitigate the effect of dynamic objects on the odometry estimation.

CRAFT [76] adopts an early fusion approach by using the mathematics of polar coordinates to associate image proposals with radar points, handling the differences in coordinate systems and spatial discrepancies. Following that, they use consecutive cross-attention feature fusion layers to exchange spatio-contextual information between the radar and camera, leading to a robust and attentive fusion.

MVFusion [77] employs a Semantic Aligned Radar Encoder (SARE) to align sparse radar features with visual semantics from multi-view camera images, producing image-guided radar representations. These enhanced radar features are then integrated with multi-scale camera features using a Radar-Guided Fusion Transformer (RGFT). The RGFT leverages a cross-attention mechanism, where the query combines radar and image features, while the key and value are derived from image features, enabling dynamic focus on relevant parts of each modality. This approach effectively learns the correlation between radar (e.g., robustness in adverse weather) and camera (e.g., rich visual context) modalities, ensuring adaptive cross-modal interaction. The resulting fused feature representation improves 3D object detection, allowing MVFusion to achieve competitive performance in detecting 3D bounding boxes on the nuScenes dataset.

CR-DINO [78] uses a Swin-transformer-based architecture [81] for sensor fusion. First, they use a unique projective transformation by labeling objects as lines and circles based on distance and shape to facilitate the projective transformation from the radar point cloud into the camera image. The image is then input into the image branch, and the projected radar points into the radar branch, which are later fused by either concatenation or addition. A transformer encoder is used to attend to the details of the representation, followed by a decoder to output the bounding boxes.

DPFT [79] is introduced to address the limitations in camera-radar fusion, particularly the sparsity of radar point clouds and the under-utilization of radar elevation data. Instead of relying on processed point clouds, DPFT leverages lower-level radar data known as the radar cube to retain more comprehensive radar information. By projecting this data onto camera and ground planes, DPFT effectively incorporates elevation information, simplifying the fusion process with camera data.

Table 5 provides a comparison of the methods, including their fusion strategies, modalities used, and the key contributions of each method.

IV. ANALYSIS AND FUTURE WORK RECOMMENDATION

A. ANALYSIS

Architectural differences between the methods lead to variations in real-time performance when deployed on a vehicle. Some architectures, such as Fusion Painting, rely on computationally expensive operations for semantic segmentation, making them less suitable for real-time deployment. Table 6 compares real-time performance for both Camera-LiDAR and Camera-Radar fusion methods, showing how architectural differences influence their performance, particularly regarding computational complexity and inference speed. Among the Camera-LiDAR methods, CMT and TransFusion stand out for their high performance. CMT achieves fast inference through a lightweight transformer architecture, while TransFusion optimizes processing with soft-association and image-guided query initialization, focusing attention on relevant features. Lift and DeepInteraction also demonstrate strong real-time capabilities due to sparse attention and efficient modality-specific representations, respectively. Conversely, Fusion Painting struggles with real-time performance due to its reliance on computationally expensive semantic segmentation, and UVTR faces moderate latency challenges due to the inherent complexity of transformers.

In the Camera-Radar category, DPFT excels in real-time efficiency by utilizing lower-level radar data and a simplified fusion approach, minimizing computational complexity. CRAFT also performs well by leveraging polar coordinate data association and cross-attention layers. TransCAR optimizes performance through sparse 3D object queries and radar feature learning. However, 4DRVO faces challenges in real-time scenarios due to the computational burden of feature pyramid networks and dynamic object handling. Moderate performers such as RCDPT, MVFusion, and CR-DINO offer valuable insights but face limitations due to architectural complexities, which affect their suitability for real-time applications.

Table 7 presents a comparison of various LiDAR-camera methods. All models were run on the same GPU, the NVIDIA V100. The training and testing were conducted using the nuScenes dataset [47]. The evaluation metrics included the nuScenes Detection Score (NDS), mean Average Precision (mAP), the reported latency of each algorithm, and the type of fusion used. The methods demonstrated very similar performance on the nuScenes dataset. Among the methods evaluated, CMT [72] achieved the highest performance based on the evaluation metrics and runtime. It's important to note that the latencies may have been measured under different configurations, which might not provide a fair basis for comparison. However, the reported latencies are included to indicate the architectural runtime cost.

The camera-radar methods are compared in Table 8. The table compares the methods based on their performance, latency, dataset, and fusion type. All algorithms have been evaluated on the nuScenes dataset, except for DPFT [79], which was evaluated on the K-Radar dataset [53]. Among the methods, DPFT [79] was the best-performing and the fastest among the camera-radar methods on the K-Radar dataset, while MV-Fusion [77] was the best-performing on the nuScenes dataset.

Table 9 shows the comparison between the best method among the camera-LiDAR methods (CMT) and the best among the camera-radar methods (DPFT). The comparison demonstrates a significant improvement of 28% for the CMT over the DPFT, as it utilizes the LiDAR sensor, which has bet-

TABLE 6. Comparison of Fusion Methods

Method	Comments on Real-Time Performance
Camera-LiDAR Fusion	
Lift [68]	Uses sparse attention for efficient processing. Suitable for real-time.
Fusion Painting [69]	Semantic segmentation increases computational cost. Not ideal for real-time.
DeepInteraction [70]	Modality-specific representations enable fast and efficient processing.
UVTR [71]	Unified representation is efficient but transformer complexity limits real-time use.
CMT [72]	Lightweight transformer enables fast inference and real-time performance.
TransFusion [67]	Uses soft-association and image-guided query initialization for robust fusion. Spatially Modulated Cross Attention (SMCA) focuses on relevant image features, reducing training time and improving detection.
Camera-Radar Fusion	
RCDPT [73]	Vision transformer is efficient, but dual fusion strategies add overhead.
TransCAR [74]	Sparse 3D object queries and radar feature learning optimize performance.
4DRVO [75]	Feature pyramid network and dynamic object handling limit real-time use.
CRAFT [76]	Polar coordinate association and cross-attention layers ensure efficiency.
MVFusion [77]	Radar-camera alignment is efficient, but multi-scale fusion adds complexity.
CR-DINO [78]	Swin-transformer is powerful but computationally intensive for real-time.
DPFT [79]	Lower-level radar data and simplified fusion enable excellent real-time performance.

TABLE 7. Comparison of LiDAR-Camera-based methods on nuScenes Dataset [47] using Nvidia V100 GPU for inference.

Method	NDS (%)	mAP (%)	Latency (ms)	Fusion Type
FusionPainting [69]	70.4	66.3	-	Deep
TransFusion [67]	71.7	68.9	265	Deep
UVTR [71]	71.1	67.1	238	Late
DeepInteraction [70]	73.4	70.8	384	Deep
Lift [68]	70.2	65.1	315	Early
CMT [72]	74.1	72.0	153	Deep

TABLE 8. Comparison of Camera-Radar-based methods

Method	NDS (%)	mAP(%)	RMSE	absRel	Latency (ms)	Dataset	Fusion type
CRAFT [76]	52.3	41.1	-	-	243 (RTX 3090)	nuScenes	Early
RCDPT [73]	-	-	5.165	0.095	111 (V100)	nuScenes	Early & Late
MVFusion [77]	51.7	45.3	-	-	-	nuScenes	Deep
TransCAR [74]	52.2	42.2	-	-	-	nuScenes	Deep
CR-Dino [78]	-	41.7	-	-	166 (RTX 4090)	nuScenes	Early
DPFT [79]	-	56.1	-	-	87 (V100)	K-Radar	Late

TABLE 9. Comparison of the best LiDAR-Camera and Radar-Camera-based methods based on mAP.

Method	mAP (%)	Modalities	Fusion Type
CMT [72]	72.0	Camera, LiDAR	Deep
DPFT [79]	56.1	Camera, Radar	Hybrid Late

ter overall depth perception than both the camera and the radar due to its higher resolution. However, LiDAR performs well only in clear or good weather conditions, and its performance deteriorates in adverse weather compared to radar. Although some datasets capture adverse weather conditions, there is no metric to reliably quantify the performance of the algorithm in extreme weather conditions.

B. FUTURE WORK RECOMMENDATION

Although the mentioned approaches did a great job addressing some of the challenges of sensor fusion, there are notable drawbacks associated with the use of transformers for this purpose. The following shows some of the existing problems and suggests some potential research avenues:

1) Adverse weather conditions: One significant challenge of transformer architectures in sensor fusion is their performance degradation in adverse weather conditions. Traditional sensors like LiDAR and cameras are known to struggle with visibility issues during rain, fog, or snow, leading to unreliable data inputs for transformers [82], [83]. The self-attention mechanisms inherent in transformers, while powerful for capturing relationships in data, can become less effective when the input data is noisy or incomplete due to environmental factors. This is particularly evident in scenarios where the performance of neural networks is heavily reliant on high-quality input data, which is often compromised in bad weather [84], [85]. Moreover, the reliance on a single type of sensor can exacerbate these issues, as each sensor modality has its limitations under specific weather conditions [86], [87].
2) Computation: Transformers also demand substantial

computational resources. The complexity of transformer models, characterized by their multi-head self-attention mechanisms, results in high computational costs, especially when processing large datasets typical in sensor fusion applications [88], [89]. This requirement can lead to increased latency in real-time applications, which is critical for autonomous driving and other time-sensitive tasks. The computational burden is further amplified when integrating multiple sensor modalities, as each modality may require separate processing pipelines before fusion can occur [90]. Consequently, the need for efficient hardware and optimized algorithms becomes paramount, which may not always be feasible in practical applications.

3) Datasets: Integrating transformers into sensor fusion frameworks often necessitates extensive training on diverse datasets to ensure robustness across varying conditions. This training process can be resource-intensive and may not guarantee performance improvements in all scenarios, particularly in unpredictable weather conditions [91]. The challenge lies in the fact that many existing datasets used for training do not adequately represent adverse weather scenarios, leading to models that perform well under ideal conditions but falter in real-world applications [84], [85]. To support advancements in Transformer-based sensor fusion, it is recommended that datasets capture a wide range of environmental conditions, including diverse weather scenarios (rain, fog, snow), lighting conditions (day, night, dusk), and dynamic environments (urban, rural, highways). Additionally, datasets should include various sensor modalities (e.g., LiDAR, radar, cameras) with accurate sensor calibration and synchronization. Publicly available datasets like KITTI, Waymo, or Apollo can serve as starting points. Collaborating with autonomous vehicle companies could help create new datasets that capture the complexities of real-world driving in challenging conditions.
4) Synchronization: Data synchronization and communication efficiency in multi-sensor networks is another critical gap. One of the main challenges to reaching a real-time latency is the transmission of large volumes of LiDAR data poses [92]. Future research could focus on developing more efficient communication protocols and data compression techniques that facilitate the timely sharing of sensor data without compromising the quality of the fused output. This could involve exploring cloud-based computation methods or decentralized fusion strategies that minimize the reliance on centralized processing to overcome heavy computation, should it become deemed [93].
5) Memory Footprint: Although transformer-based sensor fusion has a high ability to process and integrate diverse data from multiple sensors effectively, these models are often massive in terms of architecture and memory requirements [94]. Model quantization is a promising research avenue that can significantly enhance the performance of transformer-based sensor fusion systems, particularly in addressing the limitations of computational efficiency and real-time processing. By reducing the precision of model parameters and activations, quantization helps to significantly decrease memory usage and computational requirements [95]. This is especially important for deploying models in resource-limited settings, such as autonomous vehicles and mobile robotics. One of the primary advantages of model quantization is its ability to accelerate inference times. As highlighted in the literature, transformer models, while powerful, often require substantial computational resources, especially when processing high-dimensional data from multiple sensors [96], [97]. By employing quantization techniques, researchers can leverage specialized hardware that is optimized for lower-precision arithmetic, resulting in faster processing without a significant loss in accuracy. This is particularly relevant in sensor fusion applications where timely data processing is crucial for object detection and navigation [97], [98]. Moreover, quantization can help alleviate memory bandwidth constraints that arise when handling large volumes of sensor data. For instance, in scenarios where multiple sensors are providing high-resolution data streams, the ability to compress model sizes through quantization allows for more efficient data transfer between memory and processing units [98]. FrameQuant [99] outlines a simple quantization scheme to quantize transformer-based models to just two bits and some overhead, with only a small drop in accuracy, making it a good potential for deployment in real-time systems and a future avenue to investigate by quantizing existing transformer-based fusion models. This efficiency is essential for maintaining system responsiveness and reliability, especially in dynamic environments where quick decision-making is necessary.
6) Model Architecture: Another area for improvement lies in the model architecture. Recent advancements in deep learning, particularly in State-Space Models (SSMs), have proven to be very effective in modeling long sequences [100], [101]. This approach has also been adapted for image processing in the Vision Mamba architecture [102] and may similarly be adapted to sensor fusion. Additionally, research into xLSTM [103] has been proposed to address some limitations of transformers, particularly regarding robustness to noise and interpretability. Exploring the effectiveness of extreme long memory in multi-modal sensor fusion presents a promising avenue for further investigation.
7) Data Privacy and Security: Ensuring data privacy and security remains a critical challenge in federated learning (FL) for fall detection systems. While FL inherently protects user privacy by keeping data localized, there are still vulnerabilities, such as model inversion attacks and data leakage during model updates. To address

these concerns, advanced privacy-preserving techniques like homomorphic encryption and secure multi-party computation (SMPC) can be integrated into FL frameworks [104], [105]. For instance, a privacy-preserving FL framework using multi-key homomorphic encryption was proposed to ensure secure aggregation of model updates without exposing sensitive user data [106]. Similarly, FL has proven effective in non-invasive human activity recognition using channel state information [105], underscoring the importance of secure communication protocols. Future research should focus on developing robust encryption methods and secure aggregation techniques to further enhance the privacy and security of FL-based fall detection systems, ensuring compliance with regulations like GDPR [107].

8) Cloud Integration: Cloud-based sensor fusion plays a pivotal role in enhancing situational awareness and decision-making capabilities. By leveraging the computational power of cloud infrastructure, vehicles can process and analyze vast amounts of data collected from various sensors, such as LiDAR, cameras, and radar, in real time. This approach allows for the integration of information from multiple vehicles and infrastructure elements, facilitating a more comprehensive understanding of the driving environment [108], [109]. For instance, Tan et al. in [110] discuss the utilization of cloud resources for adaptive cruise control systems, where sensor data is shared between vehicles to optimize driving strategies and improve safety. Furthermore, the fusion of data in the cloud enables the application of advanced algorithms, such as machine learning and artificial intelligence, which can enhance the accuracy of object detection and tracking [13]. However, challenges remain, including the need for robust communication networks to ensure low-latency data transmission and the management of security vulnerabilities associated with data sharing in connected vehicle environments [111], [112].

To pursue the suggested future research directions, several challenges must be addressed. Adverse weather conditions pose a significant hurdle, as models need to generalize across diverse environments without extensive retraining, while balancing complexity with real-time performance [113]. Computational efficiency remains a challenge, as transformer models demand high computational resources, especially in real-time applications, requiring optimization strategies for low-latency and accuracy [114]. Synchronizing data from multiple sensors in real-time is complex, requiring efficient communication protocols and decentralized fusion strategies to minimize bottlenecks [114]. Memory footprint and large model sizes create deployment challenges in resource-constrained environments, requiring effective compression techniques [115]. Cloud integration adds complexities in latency, bandwidth, and security, further complicating scaling to large numbers of vehicles and sensors [116]. Addressing these challenges is important for advancing the field of transformer-based sensor fusion.

V. CONCLUSION

This review offers a comprehensive analysis of transformer-based sensor fusion methods across various domains, emphasizing their strengths and existing limitations. Transformer architectures have demonstrated significant potential in fusing multimodal sensor data, offering improved accuracy in complex tasks such as object detection and perception in autonomous systems. However, the computational demands, latency concerns, and reduced performance in adverse weather conditions remain key challenges that need to be addressed. Furthermore, existing datasets may not fully capture the variability in real-world scenarios, limiting the robustness of these models in practical applications. To push the boundaries of transformer-based sensor fusion, future research should focus on optimizing model architectures for efficiency, incorporating more diverse and challenging datasets, and exploring complementary fusion methods to mitigate the effects of environmental conditions. By addressing these challenges, the application of transformers in sensor fusion can unlock new opportunities for more resilient and real-time multi-sensor systems, particularly in safety-critical applications like autonomous driving and robotics.

REFERENCES

[1] J. Kabzan, M. de la Iglesia Valls, V. Reijgwart, H. F. C. Hendrikx, C. Ehmke, M. Prajapat, A. Bühler, N. B. Gosala, M. Gupta, R. Sivanesan, A. Dhall, E. Chisari, N. Karnchanachari, S. Brits, M. Dangel, I. Sa, R. Dubé, A. Gawel, M. Pfeiffer, A. Liniger, J. Lygeros, and R. Siegwart, "AMZ driverless: The full autonomous racing system," *CoRR*, vol. abs/1905.05150, 2019.

[2] McKinsey & Company, "Autonomous vehicles: Moving forward—perspectives from industry leaders," McKinsey Center for Future Mobility, 2021, accessed: April 22, 2024. [Online]. Available: https://www.mckinsey.com/features/mckinsey-center-for-future-mobility/our-insights/autonomous-vehicles-moving-forward-perspectives-from-industry-leaders

[3] J. Fayyad, M. A. Jaradat, D. Gruyer, and H. Najjaran, "Deep learning sensor fusion for autonomous vehicle perception and localization: A review," *Sensors*, vol. 20, no. 15, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/15/4220

[4] Society of Automotive Engineers. (2018) Automatic emergency braking (aeb) system performance testing. Accessed: April 15, 2024. [Online]. Available: https://saemobilus.sae.org/content/j3016$_2$01806

[5] C. Wang, L. Zhi, and Y. Wang, "Intelligent fault diagnosis of photoelectric pod bearing based on multi-information fusion," *Journal of Physics: Conference Series*, vol. 2136, p. 012036, 2021.

[6] A. Singh, "Transformer-based sensor fusion for autonomous driving: A survey," in *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2023, pp. 3304–3309.

[7] D. J. Yeong, G. Velasco-Hernandez, J. Barry, and J. Walsh, "Sensor and sensor fusion technology in autonomous vehicles: A review," *Sensors*, vol. 21, no. 6, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/6/2140

[8] Intel. (2023) Buy intel realsense depth camera d455. Accessed: January 25, 2025. [Online]. Available: https://store.intelrealsense.com/buy-intel-realsense-depth-camera-d455.html

[9] Neuvition. (2023) Lidar price: How much does lidar cost? Accessed: January 25, 2025. [Online]. Available: https://www.neuvition.com/media/blog/lidar-price.html

[10] Conti Engineering. (2023) Ars-408 radar sensor. Accessed: January 25, 2025. [Online]. Available: https://conti-engineering.com/components/ars-408/

[11] B. Khaleghi, A. Khamis, F. O. Karray, and S. N. Razavi, "Multisensor data fusion: A review of the state-of-the-art," *Information Fusion*, vol. 14, no. 1, pp. 28–44, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1566253511000558

[12] D. Jiang, D. Zhuang, Y. Huang, and J. Fu, "Advances in multi-sensor data fusion: algorithms and applications," *Sensors*, vol. 9, pp. 7771–7784, 2009.

[13] J. Fayyad, M. Jaradat, D. Gruyer, and H. Najjaran, "Deep learning sensor fusion for autonomous vehicle perception and localization: a review," *Sensors*, vol. 20, p. 4220, 2020.

[14] F. Nobis, M. Geisslinger, M. Weber, J. Betz, and M. Lienkamp, "A deep learning-based radar and camera sensor fusion architecture for object detection," 2019.

[15] V. John and S. Mita, "Deep feature-level sensor fusion using skip connections for real-time object detection in autonomous driving," *Electronics*, vol. 10, p. 424, 2021.

[16] E. Cinar, "A sensor fusion method using transfer learning models for equipment condition monitoring," *Sensors*, vol. 22, p. 6791, 2022.

[17] M. Bihler, "Multi-sensor data fusion using deep learning for bulky waste image classification," 2023.

[18] T. Baltrušaitis, C. Ahuja, and L. Morency, "Multimodal machine learning: a survey and taxonomy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, pp. 423–443, 2019.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2023.

[20] A. Botev, S. De, S. L. Smith, A. Fernando, G.-C. Muraru, R. Haroun, L. Berrada, R. Pascanu, P. G. Sessa, R. Dadashi, L. Hussenot, J. Ferret, S. Girgin, O. Bachem, A. Andreev, K. Kenealy, T. Mesnard, C. Hardin, S. Bhupatiraju, S. Pathak, L. Sifre, M. Rivière, M. S. Kale, J. Love, P. Tafti, A. Joulin, N. Fiedel, E. Senter, Y. Chen, S. Srinivasan, G. Desjardins, D. Budden, A. Doucet, S. Vikram, A. Paszke, T. Gale, S. Borgeaud, C. Chen, A. Brock, A. Paterson, J. Brennan, M. Risdal, R. Gundluru, N. Devanathan, P. Mooney, N. Chauhan, P. Culliton, L. G. Martins, E. Bandy, D. Huntsperger, G. Cameron, A. Zucker, T. Warkentin, L. Peran, M. Giang, Z. Ghahramani, C. Farabet, K. Kavukcuoglu, D. Hassabis, R. Hadsell, Y. W. Teh, and N. de Frietas, "Recurrentgemma: Moving past transformers for efficient open language models," 2024.

[21] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, "itransformer: Inverted transformers are effective for time series forecasting," 2024.

[22] S. Latif, A. Zaidi, H. Cuayahuitl, F. Shamshad, M. Shoukat, and J. Qadir, "Transformers in speech processing: A survey," 2023.

[23] 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," *CoRR*, vol. abs/2010.11929, 2020.

[24] Z. Zhang, M. Farnsworth, B. Song, D. Tiwari, and A. Tiwari, "Deep transfer learning with self-attention for industry sensor fusion tasks," *Ieee Sensors Journal*, vol. 22, pp. 15 235–15 247, 2022.

[25] S. Alaba, "A comprehensive survey of deep learning multisensor fusion-based 3d object detection for autonomous driving: methods, challenges, open issues, and future directions," 2022.

[26] Z. Huang, "Multi-modal sensor fusion-based deep neural network for end-to-end autonomous driving with scene understanding," 2020.

[27] D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Gläser, F. Timm, W. Wiesbeck, and K. Dietmayer, "Deep multi-modal object detection and semantic segmentation for autonomous driving: datasets, methods, and challenges," *Ieee Transactions on Intelligent Transportation Systems*, vol. 22, pp. 1341–1360, 2021.

[28] A. Abdullahi, "Lane tracking in self-driving cars: leveraging tensorflow for deep learning in image processing across localization, and sensor fusion," *Advances in Computer and Communication*, vol. 4, pp. 271–276, 2023.

[29] Z. Wang, Y. Wu, and Q. Niu, "Multi-sensor fusion in automated driving: a survey," *Ieee Access*, vol. 8, pp. 2847–2868, 2020.

[30] L. Torbarina, T. Ferkovic, L. Roguski, V. Mihelcic, B. Sarlija, and Z. Kraljevic, "Challenges and opportunities of using transformer-based multi-task learning in nlp through ml lifecycle: A survey," 2023. [Online]. Available: https://arxiv.org/abs/2308.08234

[31] X. Wang, K. Li, and A. Chehri, "Multi-sensor fusion technology for 3d object detection in autonomous driving: A review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 2, pp. 1148–1165, 2024.

[32] K. Huang, B. Shi, X. Li, X. Li, S. Huang, and Y. Li, "Multi-modal sensor fusion for auto driving perception: A survey," 2022. [Online]. Available: https://arxiv.org/abs/2202.02703

[33] D. J. Yeong, G. Velasco-Hernandez, J. Barry, and J. Walsh, "Sensor and sensor fusion technology in autonomous vehicles: A review," *Sensors*, vol. 21, no. 6, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/6/2140

[34] H. Kuang, X. Liu, J. Zhang, and Z. Fang, "Multi-modality cascaded fusion technology for autonomous driving," *2020 4th International Conference on Robotics and Automation Sciences (ICRAS)*, 2020.

[35] D. Xu, D. Anguelov, and A. Jain, "Pointfusion: deep sensor fusion for 3d bounding box estimation," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[36] M. Liang, B. Yang, Y. Chen, R. Hu, and R. Urtasun, "Multi-task multi-sensor fusion for 3d object detection," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[37] T. Liang, H. Xie, K. Yu, Z. Xia, Z. Lin, Y. Wang, T. Tang, B. Wang, and Z. Tang, "Bevfusion: A simple and robust lidar-camera fusion framework," 2022.

[38] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," *CoRR*, vol. abs/2008.05711, 2020.

[39] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," *CoRR*, vol. abs/1709.01507, 2017.

[40] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," *CoRR*, vol. abs/1611.07759, 2016.

[41] K. Chitta, A. Prakash, B. Jaeger, Z. Yu, K. Renz, and A. Geiger, "Transfuser: Imitation with transformer-based sensor fusion for autonomous driving," 2022.

[42] A. J. Piergiovanni, V. Casser, M. S. Ryoo, and A. Angelova, "4d-net for learned multi-modal alignment," *CoRR*, vol. abs/2109.01066, 2021.

[43] S. Pang, D. Morris, and H. Radha, "Clocs: Camera-lidar object candidates fusion for 3d object detection," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 10 386–10 393.

[44] A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret, "Transformers are rnns: Fast autoregressive transformers with linear attention," *CoRR*, vol. abs/2006.16236, 2020. [Online]. Available: https://arxiv.org/abs/2006.16236

[45] 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," *CoRR*, vol. abs/2010.11929, 2020. [Online]. Available: https://arxiv.org/abs/2010.11929

[46] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019. [Online]. Available: https://arxiv.org/abs/1810.04805

[47] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," *CoRR*, vol. abs/1903.11027, 2019. [Online]. Available: http://arxiv.org/abs/1903.11027

[48] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[49] T. Gruber, F. Julca-Aguilar, M. Bijelic, and F. Heide, "Gated2depth: Real-time dense lidar from gated images," in *The IEEE International Conference on Computer Vision (ICCV)*, 2019.

[50] X. Huang, X. Cheng, Q. Geng, B. Cao, D. Zhou, P. Wang, Y. Lin, and R. Yang, "The apolloscape dataset for autonomous driving," *CoRR*, vol. abs/1803.06184, 2018. [Online]. Available: http://arxiv.org/abs/1803.06184

[51] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," 2016. [Online]. Available: https://arxiv.org/abs/1604.01685

[52] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, S. Zhao, S. Cheng, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov, "Scalability in perception for autonomous driving: Waymo open dataset," 2020. [Online]. Available: https://arxiv.org/abs/1912.04838

[53] D.-H. Paek, S.-H. Kong, and K. T. Wijaya, "K-radar: 4d radar object detection for autonomous driving in various weather conditions," 2023. [Online]. Available: https://arxiv.org/abs/2206.08171

[54] J. Li, R. Xu, X. Liu, J. Ma, B. Li, Q. Zou, J. Ma, and H. Yu, "Domain adaptation based object detection for autonomous driving in foggy and rainy weather," *IEEE Transactions on Intelligent Vehicles*, pp. 1–12, 2024.

[55] A. Mauri, R. Khemmar, B. Decoux, M. Haddad, and R. Boutteau, "Lightweight convolutional neural network for real-time 3d object detection in road and railway environments," *Journal of Real-Time Image Processing*, 2022.

[56] A. Srivastav, "Radars for autonomous driving: A review of deep learning methods and challenges," *Ieee Access*, 2023.

[57] H. Rashed, M. Ramzy, V. Vaquero, A. E. Sallab, G. Sistu, and S. Yogamani, "Fusemodnet: Real-time camera and lidar based moving object detection for robust low-light autonomous driving," 2019.

[58] S. Alaba and J. Ball, "Multi-sensor fusion 3d object detection for autonomous driving," 2023.

[59] D. Xu, D. Anguelov, and A. Jain, "Pointfusion: Deep sensor fusion for 3d bounding box estimation," 2018.

[60] P. Kowalczyk, M. Komorkiewicz, P. Skruch, and M. Szelest, "Efficient characterization method for big automotive datasets used for perception system development and verification," *Ieee Access*, 2022.

[61] H. Kuang, X. Liu, J. Zhang, and Z. Fang, "Multi-modality cascaded fusion technology for autonomous driving," 2020.

[62] M. Adam, M. Piccolrovazzi, S. Eger, and E. Steinbach, "Bounding box disparity: 3d metrics for object detection with full degree of freedom," 2022.

[63] X. Chen, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," 2017.

[64] M. Liang, B. Yang, Y. Chen, R. Hu, and R. Urtasun, "Multi-task multi-sensor fusion for 3d object detection," 2019.

[65] O. J. Montañez, "Application of data sensor fusion using extended kalman filter algorithm for identification and tracking of moving targets from lidar–radar data," *Remote Sensing*, 2023.

[66] D. Zhou, F. Jin, X. Song, C. Guan, J. Yin, Y. Dai, and R. Yang, "Iou loss for 2d/3d object detection," 2019.

[67] X. Bai, Z. Hu, X. Zhu, Q. Huang, Y. Chen, H. Fu, and C.-L. Tai, "Transfusion: Robust lidar-camera fusion for 3d object detection with transformers," 2022.

[68] Y. Zeng, D. Zhang, C. Wang, Z. Miao, T. Liu, X. Zhan, D. Hao, and C. Ma, "Lift: Learning 4d lidar image fusion transformer for 3d object detection," 06 2022, pp. 17 151–17 160.

[69] S. Xu, D. Zhou, J. Fang, J. Yin, Z. Bin, and L. Zhang, "Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection," 2021. [Online]. Available: https://arxiv.org/abs/2106.12449

[70] Z. Yang, J. Chen, Z. Miao, W. Li, X. Zhu, and L. Zhang, "Deepinteraction: 3d object detection via modality interaction," 2022. [Online]. Available: https://arxiv.org/abs/2208.11112

[71] Y. Li, Y. Chen, X. Qi, Z. Li, J. Sun, and J. Jia, "Unifying voxel-based representation with transformer for 3d object detection," 2022. [Online]. Available: https://arxiv.org/abs/2206.00630

[72] J. Yan, Y. Liu, J. Sun, F. Jia, S. Li, T. Wang, and X. Zhang, "Cross modal transformer: Towards fast and robust 3d object detection," 2023. [Online]. Available: https://arxiv.org/abs/2301.01283

[73] C.-C. Lo and P. Vandewalle, "Rcdpt: Radar-camera fusion dense prediction transformer," Piscataway, NJ, USA, 2023//, pp. 1 – 5. [Online]. Available: http://dx.doi.org/10.1109/ICASSP49357.2023.10096129

[74] S. Pang, D. Morris, and H. Radha, "Transcar: Transformer-based camera-and-radar fusion for 3d object detection," Piscataway, NJ, USA, 2023//, pp. 10 902 – 9. [Online]. Available: http://dx.doi.org/10.1109/IROS55552.2023.10341793

[75] S. Lu, G. Zhuo, L. Xiong, M. Zhou, and X. Lu, "Deep 4d automotive radar-camera fusion odometry with cross-modal transformer fusion," no. 357463, USA, 2023//, pp. 2023 – 01. [Online]. Available: http://dx.doi.org/10.4271/2023-01-7040

[76] Y. Kim, S. Kim, J. W. Choi, and D. Kum, "Craft: Camera-radar 3d object detection with spatio-contextual fusion transformer," vol. 37, Washington, DC, United states, 2023, pp. 1160 – 1168.

[77] Z. Wu, G. Chen, Y. Gan, L. Wang, and J. Pu, "Mvfusion: Multi-view 3d object detection with semantic-aligned radar and camera fusion," Piscataway, NJ, USA, 2023//, pp. 2766 – 73. [Online]. Available: http://dx.doi.org/10.1109/ICRA48891.2023.10161329

[78] Y. Jin, X. Zhu, Y. Yue, E. Lim, and W. Wang, "Cr-dino: A novel camera-radar fusion 2-d object detection model based on transformer," *IEEE Sensors Journal*, vol. 24, no. 7, pp. 11 080 – 90, 2024//. [Online]. Available: http://dx.doi.org/10.1109/JSEN.2024.3357775

[79] F. Fent, A. Palffy, and H. Caesar, "Dpft: Dual perspective fusion transformer for camera-radar-based object detection," 2024/04/03. [Online]. Available: http://dx.doi.org/10.48550/arXiv.2404.03015

[80] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," 2021.

[81] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," 2021. [Online]. Available: https://arxiv.org/abs/2103.14030

[82] H. Li, N. Bamminger, Z. F. Magosi, C. Feichtinger, Y. Zhao, T. Mihalj, F. Orucevic, and A. Eichberger, "The effect of rainfall and illumination on automotive sensors detection performance," *Sustainability*, vol. 15, no. 9, 2023. [Online]. Available: https://www.mdpi.com/2071-1050/15/9/7260

[83] S. S. Chaturvedi, L. Zhang, and X. Yuan, "Pay "attention" to adverse weather: Weather-aware attention-based object detection," 2022. [Online]. Available: https://arxiv.org/abs/2204.10803

[84] A. Pfeuffer and K. Dietmayer, "Optimal sensor data fusion architecture for object detection in adverse weather conditions," in *2018 21st International Conference on Information Fusion (FUSION)*, 2018, pp. 1–8.

[85] ——, "Robust semantic segmentation in adverse weather conditions by means of sensor data fusion," 2019. [Online]. Available: https://arxiv.org/abs/1905.10117

[86] C. Pereira, R. P. M. Cruz, J. N. D. Fernandes, J. R. Pinto, and J. S. Cardoso, "Weather and meteorological optical range classification for autonomous driving," *IEEE Transactions on Intelligent Vehicles*, pp. 1–11, 2024.

[87] A. R. Abbasi, "Comparison parametric and non-parametric methods in probabilistic load flow studies for power distribution networks," *Electrical Engineering*, vol. 104, no. 6, pp. 3943–3954, Dec 2022. [Online]. Available: https://doi.org/10.1007/s00202-022-01590-9

[88] S. Y. Alaba and J. E. Ball, "Transformer-based optimized multimodal fusion for 3d object detection in autonomous driving," *IEEE Access*, vol. 12, pp. 50 165–50 176, 2024.

[89] L. Wen, Y. Peng, M. Lin, N. Gan, and R. Tan, "Multi-modal contrastive learning for lidar point cloud rail-obstacle detection in complex weather," *Electronics*, vol. 13, no. 1, 2024. [Online]. Available: https://www.mdpi.com/2079-9292/13/1/220

[90] J.-H. Choi, K.-B. Kang, and K.-T. Kim, "Fusion-vital: Video-rf fusion transformer for advanced remote physiological measurement," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 2, pp. 1344–1352, Mar. 2024. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/27898

[91] S. Alaba, A. Gürbüz, and J. Ball, "A comprehensive survey of deep learning multisensor fusion-based 3d object detection for autonomous driving: methods, challenges, open issues, and future directions," 2022.

[92] Q. Chen, S. Tang, Q. Yang, and S. Fu, "Cooper: cooperative perception for connected autonomous vehicles based on 3d point clouds," 2019.

[93] W. Liu, X. Zang, Y. Li, and B. Vucetic, "Over-the-air computation systems: optimization, analysis and scaling laws," 2019.

[94] L. Sun, G. Ding, Y. Yoshiyasu, and F. Kanehiro, "Certainodom: uncertainty weighted multi-task learning model for lidar odometry estimation," *2022 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2022.

[95] S. Shen, D. Zhen, J. Ye, L. Ma, Z. Yao, A. Gholami, M. Mahoney, and K. Keutzer, "Q-bert: Hessian based ultra low precision quantization of bert," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 8815–8821, 04 2020.

[96] J. Gu, A. Lind, T. R. Chhetri, M. Bellone, and R. Sell, "End-to-end multimodal sensor dataset collection framework for autonomous vehicles," *Sensors*, vol. 23, p. 6783, 2023.

[97] K. R. Shahi, P. Ghamisi, B. Rasti, R. Jackisch, P. Scheunders, and R. Gloaguen, "Data fusion using a multi-sensor sparse-based clustering algorithm," *Remote Sensing*, vol. 12, p. 4007, 2020.

[98] W. Wu and C. Liu, "Research on traffic object recognition based on multi-sensor fusion," *International Conference on Computer Graphics, Artificial Intelligence, and Data Processing (ICCAID 2023)*, 2024.

[99] H. Adepu, Z. Zeng, L. Zhang, and V. Singh, "Framequant: Flexible low-bit quantization for transformers," 2024. [Online]. Available: https://arxiv.org/abs/2403.06082

[100] X. Wang, S. Wang, Y. Ding, Y. Li, W. Wu, Y. Rong, W. Kong, J. Huang, S. Li, H. Yang, Z. Wang, B. Jiang, C. Li, Y. Wang, Y. Tian, and J. Tang, "State space model for new-generation network alternative to transformers: A survey," 2024. [Online]. Available: https://arxiv.org/abs/2404.09516

[101] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," 2024. [Online]. Available: https://arxiv.org/abs/2312.00752
[102] L. Zhu, B. Liao, Q. Zhang, X. Wang, W. Liu, and X. Wang, "Vision mamba: Efficient visual representation learning with bidirectional state space model," 2024. [Online]. Available: https://arxiv.org/abs/2401.09417
[103] M. Beck, K. Pöppel, M. Spanring, A. Auer, O. Prudnikova, M. Kopp, G. Klambauer, J. Brandstetter, and S. Hochreiter, "xlstm: Extended long short-term memory," 2024. [Online]. Available: https://arxiv.org/abs/2405.04517
[104] J. Ma, S. Naas, S. Sigg, and X. Lyu, "Privacy-preserving federated learning based on multi-key homomorphic encryption," *CoRR*, vol. abs/2104.06824, 2021. [Online]. Available: https://arxiv.org/abs/2104.06824
[105] P. Qi, D. Chiaro, and F. Piccialli, "Fl-fd: Federated learning-based fall detection with multimodal data fusion," *Information Fusion*, vol. 99, p. 101890, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1566253523002063
[106] V. P. Chellapandi, L. Yuan, C. G. Brinton, S. H. Zak, and Z. Wang, "Federated learning for connected and automated vehicles: A survey of existing approaches and challenges," 2023. [Online]. Available: https://arxiv.org/abs/2308.10407
[107] H. Huang, "Defense against membership inference attack applying domain adaptation with addictive noise," *J. Comput. Commun.*, vol. 09, no. 05, pp. 92–108, 2021.
[108] Z. Tang, J. He, S. Flanagan, P. Procter, and L. Cheng, "Cooperative connected smart road infrastructure and autonomous vehicles for safe driving," pp. 1–6, 2021.
[109] F. Butt, J. Chattha, J. Ahmad, U. Zia, M. Rizwan, and I. Naqvi, "On the integration of enabling wireless technologies and sensor fusion for next-generation connected and autonomous vehicles," *Ieee Access*, vol. 10, pp. 14 643–14 668, 2022.
[110] M. Tan, M. Li, and Q. Abbasi, "Sensor aided beamforming in vehicular environment," 2020.
[111] J. Ahmad, "Machine learning and blockchain technologies for cybersecurity in connected vehicles," *Wiley Interdisciplinary Reviews Data Mining and Knowledge Discovery*, vol. 14, 2023.
[112] M. Xi, D. Duan, and T. Feng, "Multi-vehicle multi-sensor occupancy grid map fusion in vehicular networks," *Iet Communications*, vol. 16, pp. 67–74, 2021.
[113] C. Xu and R. Sankar, "A comprehensive review of autonomous driving algorithms: Tackling adverse weather conditions, unpredictable traffic violations, blind spot monitoring, and emergency maneuvers," *Algorithms*, vol. 17, no. 11, 2024, adverse weather;Adverse weather condition;Autonomous driving;Autonomous driving algorithm;Condition;Learning optimizations;Machine learning optimization;Machine-learning;Real time decision-making;Real-time decision making;Safety and reliability;Sensor fusion;. [Online]. Available: http://dx.doi.org/10.3390/a17110526
[114] Y.-L. Tian, Y.-T. Wang, J.-G. Wang, X. Wang, and F.-Y. Wang, "Key problems and progress of vision transformers: The state of the art and prospects," *Zidonghua Xuebao/Acta Automatica Sinica*, vol. 48, no. 4, pp. 957 – 979, 2022, computing capability;Convolutional neural network;Dynamic weight;Existing problems;Images classification;Images segmentations;Parallel computing;Sequence models;State of the art;Vision transformer;. [Online]. Available: http://dx.doi.org/10.16383/j.aas.c220027
[115] Z. Lu, F. Wang, Z. Xu, F. Yang, and T. Li, "On the performance and memory footprint of distributed training: An empirical study on transformers," 2024. [Online]. Available: https://arxiv.org/abs/2407.02081
[116] P. Arthurs, L. Gillam, P. Krause, N. Wang, K. Halder, and A. Mouzakitis, "A taxonomy and survey of edge cloud computing for intelligent transportation systems and connected vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 6206–6221, 2022.

AHMED ABDULMAKSOUD received the B.Sc. degree in electrical engineering from Ain Shams University, Cairo, Egypt, in 2023. He is currently pursuing the M.A.Sc. degree in mechanical engineering at McMaster University, Hamilton, ON, Canada, and working as an Intern Machine Learning Developer at Kinaxis. In 2024, Ahmed was a Development Intern at Stellantis, Windsor, ON, Canada. From 2022 to 2023, he served as a Software Engineer Intern at Siemens Digital Industry Software in Cairo, Egypt. During this time, he also led autonomous perception for the ASU Racing Team's Autotronics Research Lab, focusing on the Formula Student AI Competition in the UK, where he worked on autonomous racing cars and drones for environment mapping. His research interests include machine learning applications in robotics, electric vehicles, autonomous perception, 3D scanning, and the use of cloud and distributed software systems across various engineering fields.

RYAN AHMED Dr. Ryan Ahmed is an Assistant Professor at McMaster University, acting deputy director of the Center for Mechatronics and Hybrid Technologies (CMHT), and co-lead faculty advisor for the Battery Workforce Challenge (BWC). He received his M.A.Sc., Ph.D., and MBA from McMaster University in 2011, 2014, and 2018 respectively. He has held several senior positions in Electric and Autonomous vehicles at General Motors, Samsung, and Stellantis in Canada and the United States. Dr. Ahmed has taught over half a million learners from 160 countries on Udemy and Coursera, and he has over 250,000 subscribers on his YouTube channel titled "Prof. Ryan Ahmed," where he teaches people AI, data science, and ML fundamentals. Dr. Ahmed is a Udemy Instructor Partner, Professional Engineer (P.Eng.) in Ontario, and Stanford Certified Program Manager. He is the principal author/co-author of over 45 journal and conference papers in artificial intelligence, battery systems, electric and hybrid powertrains, and autonomous systems. Dr. Ahmed is an associate editor at the IEEE Transactions on Transportation Electrification journal. He is the co-recipient of two best papers awards at the IEEE Transactions on Industrial Electronics (2018) and the IEEE Transportation Electrification Conference and Expo (ITEC 2012) in Detroit, MI, USA.