IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 22 December 2023, accepted 1 April 2024, date of publication 5 April 2024, date of current version 12 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3385439

APPLIED RESEARCH

Transformer-Based Optimized Multimodal Fusion for 3D Object Detection in Autonomous Driving

SIMEGNEW YIHUNIE ALABA, (Member, IEEE), AND JOHN E. BALL, (Senior Member, IEEE)
Department of Electrical and Computer Engineering, James Worth Bagley College of Engineering, Mississippi State University, Starkville, MS 39762, USA
Corresponding author: Simegnew Yihunie Alaba (sa1724@msstate.edu)

ABSTRACT Accurate 3D object detection is vital for autonomous driving since it facilitates accurate perception of the environment through multiple sensors. Although cameras can capture detailed color and texture features, they have limitations regarding depth information. Additionally, they can struggle under adverse weather or lighting conditions. In contrast, LiDAR sensors offer robust depth information but lack the visual detail for precise object classification. This work presents a multimodal fusion model that improves 3D object detection by combining the benefits of LiDAR and camera sensors to address these challenges. This model processes camera images and LiDAR point cloud data into a voxel-based representation, further refined by encoder networks to enhance spatial interaction and reduce semantic ambiguity. The proposed multiresolution attention module and integration of discrete wavelet transform and inverse discrete wavelet transform to the image backbone improve the feature extraction capability. This approach enhances the fusion of LiDAR depth information with the camera's textural and color detail. The model also incorporates a transformer decoder network with self-attention and cross-attention mechanisms, fostering robust and accurate detection through global interaction between identified objects and encoder features. Furthermore, the proposed network is refined with advanced optimization techniques, including pruning and Quantization-Aware Training (QAT), to maintain a competitive performance while significantly decreasing the need for memory and computational resources. Performance evaluations on the nuScenes dataset show that the optimized model architecture offers competitive results and significantly improves operational efficiency and effectiveness in multimodal fusion 3D object detection.

INDEX TERMS Autonomous driving, LiDAR, multimodal fusion, network compression, pruning, quantization, quantization-aware training, sparsity, vision transformer, 3D object detection.

I. INTRODUCTION

Three-dimensional object detection is becoming more popular recently due to the rapid growth of deep learning models, and the motivation to know precise object localization has been increased, especially in robotics applications, such as autonomous driving. Different sensors, such as cameras and LiDARs, are available for such applications. Although cameras provide rich color and texture information, their effectiveness in 3D detection is limited by their lack of 3D information and susceptibility to inclement weather. LiDARs, on the other hand, offer 3D data appropriate for applications involving three dimensions. Besides their sparsity, LiDARs lack color information, and texture information is harder to extract from a LiDAR than from a camera. Hence, no sensor is suitable for every application, and sensors have limitations [1]. Multimodal fusion is frequently used in autonomous driving to lessen the impact of sensor constraints, extract complementary knowledge from other sensors—such as color from the camera and 3D information from LiDAR and use redundant data in the event of sensor failure.

As various sensors have different input representations, fusing features from multiple sensors is challenging. The integration of cameras and LiDAR is frequently used for 3D object detection. There are various techniques to reduce

The associate editor coordinating the review of this manuscript and approving it for publication was Emre Koyuncu.

the impact of input mismatch during multimodal fusion. The first way is to project LiDAR data into a 2D representation and merge it with related camera data, such as range image or bird's-eye view (BEV) [2], [3], [4]. The second approach, as seen in [5], involves transforming camera data into a pseudo-LiDAR representation and fusing them into a LiDAR representation. Information is lost when projecting the LiDAR data into a 2D representation. On the other hand, errors can also occur while converting a camera image to a pseudo-LiDAR representation [1]. Another method is to transform camera images into voxels and fuse them with LiDAR voxel data [6]. Furthermore, no optimal or widely accepted method for combining multimodal data exists. Projecting LiDAR data into the BEV representation also causes height compression, which may lead to semantic ambiguity. This work constructs a voxel space from images based on predicted depth scores and geometric constraints without height compression. This representation helps to associate with the corresponding LiDAR point cloud features. The voxel encoder network is used for spatial interaction between the constructed voxel and the LiDAR point cloud features. The final transformer decoder network is the detection head and samples particular features for each object's 3D query coordinates.

Furthermore, a multiresolution convolutional attention module is proposed to enhance the network's feature extraction capability. This module leverages the multiresolution characteristics of wavelets to extract features at multiple scales, highlighting the most significant features. The design of this attention module leverages the inherent properties of wavelets, such as sparsity and the ability to separate noise from the image. Wavelets are highly effective at capturing noise and isolating it within the high-frequency component [7]. This property allows for discarding much of the noise component during processing attention weights (see Section III). Additionally, given the inherent characteristics of wavelets, the discrete wavelet transform (DWT) and inverse discrete wavelet transform (IDWT) have been incorporated into the BasicBlock and Bottleneck blocks of the image backbone ResNet [8] network to enhance the network's feature extraction capability.

Although multimodal fusion improves detection and localization performance, the computational burden is a large deployment obstacle in resource-constrained environments. The problem is more pronounced when the transformer network is used because of the computational burden of the multihead self-attention module. This work employs pruning and quantization to reduce computational burden and optimize the model for faster training and inference. Network pruning removes weight parameters that do not affect the model's performance. On the other hand, network quantization involves reducing the bit representation of the data, for example, using eight-bit integers (INT8) instead of floating point 32-bit (FP32) [9]. Finally, quantization-aware training (QAT) is employed to mitigate the accuracy loss during quantization.

The main contributions can be summarized as follows:

1) A Multimodal fusion model for 3-D object detection has been developed, utilizing voxels as a unified representation.
2) A Transformer decoder is utilized to enhance detection performance by leveraging self-attention and cross-attention mechanisms for long-range dependencies.
3) A multiresolution-based convolutional attention mechanism to improve the feature extraction capabilities of the backbone network is proposed and evaluated.
4) A new backbone network is proposed by integrating DWT and IDWT with the BasicBlock and Bottleneck blocks of the ResNet backbone network to enhance the network's feature extraction capabilities.
5) Implement pruning and QAT techniques for network optimization to reduce computational load and enhance performance.
6) Multiple experiments were conducted to demonstrate the performance of the proposed model on the nuScenes dataset.

The structure of the rest of the paper is as follows: Section II reviews work related to the proposed work, and Section III details the architecture of the proposed model, including feature extractors, unified feature representation, transformer decoders, multiresolution attention module, and network optimization. Section IV is dedicated to the experimental results and analyses, including the setup of the training process. Finally, Section V provides a conclusion on the experimental results of the proposed model.

II. RELATED WORK

A. CAMERA-BASED 3D OBJECT DETECTION

The lack of accurate depth information is challenging for camera-based 3D object detection. Various methods have been developed for estimating depth from stereo or monocular images. Among these, one technique for estimating depth is to use LiDAR-based models, such [10], [11], and convert the camera image into a pseudo-LiDAR representation. Some approaches, including [12], [13], [14], estimate depth using stereo images to address the lack of depth issue. Geometric constraints, such as the object's shape and key points like [15], [16], and [17], can also solve the lack of depth. Depth camera-based estimate techniques are essential when 3D sensors are difficult to obtain, even though they have distinct associated issues and are less accurate than equivalent 3D sensors like LiDAR. Furthermore, cameras are widely accessible due to their lower cost than most 3D sensors.

B. LIDAR-BASED 3D OBJECT DETECTION

Point cloud data is unstructured and sparse, which makes LiDAR processing more challenging [18]. Various methods have been developed for processing and encoding LiDAR data. These techniques can be grouped into three categories: voxel, projection, and raw point cloud. The Projection of LiDAR data into a 2D representation employs regression

approaches and 2D methods to produce 3D information at the final detection stages. For instance, BirdNet+ [19] used a bird's-eye view (BEV) to project the LiDAR point cloud and then performed 2D convolution on the projected data. Voxel methods, including VoxelNet [20], create volumetric (voxel) representations from stacked point clouds to create LiDAR point clouds. SECOND [21] enhanced voxelNet and introduced sparse convolution [22]. Point cloud data can also be encoded into pillar representations, such as PointPillars [23] and WCNN3D [24]. The pillar representation through 2D convolution can alleviate the computational load of 3D convolution on a voxel representation. Information loss can occur when the LiDAR data are sampled into a voxel grid or projected onto a 2D representation. Processing the raw LiDAR data prevents information loss. However, in addition to the processing overhead resulting from 3D convolution, another difficulty is the sparse and unstructured nature of point-cloud data. PointNet [25] and PointNet++ [26] are widely used 3D detection methods that directly process LiDAR data. Some works integrate the raw point cloud and the voxel grid, such as PV-RCNN [27]. The model outperformed voxel-only techniques in terms of performance, but its computational complexity makes it impossible to use in real-time.

C. MULTIMODAL 3D OBJECT DETECTION

Multimodal fusion uses multiple inputs from various sensors for robust detection. Although multimodal fusion-based 3D object detection methods outperform LiDAR-only or camera-only methods, the data mismatch due to different input representations is challenging. The multimodal fusion methods can be categorized into data, feature, or decision fusion. Data fusion, sometimes called early fusion, fuses raw data at the beginning of the network, making it challenging due to the difference in the representation of the input data [28]. AVOD [3] fuses LiDAR and camera data at the beginning of the network. In decision fusion, sometimes called late fusion, features from multiple sensors are processed separately and transformed into high-level features before the fusion point. So, this fusion type does not have a data mismatch problem like the data fusion. Wang et al. [29] proposed a Voxel-Pixel fusion network that fuses LiDAR and image at the voxel level. The image features are voxelized before fusing with the corresponding LiDAR voxelized features. The inputs from various sensors are processed separately until the fusion layer for the decision fusion. This may restrict the proper use of input from different sensors. Feature fusion may solve the problems of data and decision fusions. Feature fusion can be a middle-level, one-layer, or deep fusion, so it does not have a data mismatch problem [28]. It correctly uses all the data up to the decision level. PointPainting [30] fuses LiDAR features and camera image features. The pointPainting performance depends on the segmentation performance. Therefore, it fails for small objects as a result of segmentation failure. Although feature fusion uses fused data properly, data or decision fusion may outperform it. Due to this and other related issues, the best and agreed fusion technique is not yet solved [28]. Similarly, PointAugmenting [31] decorates the point cloud data with CNN 2D image pointwise features to solve the related segmentation issues of the pointPainting [30].

Transformer networks have shown success in sequence-to-sequence tasks. Recently, transformers have also succeeded in different vision applications, including object detection. Different transformer-based 3D fusion detection models have been proposed [4], [6], [32], [33], [34], [35], [36], [37]. UVTR [6] and Multimodal fusion [36] networks unified multimodal inputs of camera images and LiDAR point cloud using a transformer network. The image data are transformed into voxel space, and the LiDAR point cloud is sampled into voxel space before fusion. In contrast, BEVFusion [37] is a multitask multisensor model that fuses the LiDAR point cloud and camera image data in a BEV representation. Token-Fusion [32] is a multimodal fusion network that dynamically detects tokens and substitutes with projected intermodal features. Residual positional alignment is also used to utilize intermodal alignments better. Bridge-transformer network [33] is a 3D object detection network designed to bridge the learning process of image and point cloud data. Conditional object queries and points-to-patch projection are proposed to increase the interaction between image and point cloud data. DeepFusion [35] put forth a multimodal fusion with InverseAug and LearnableAlign to enable geometric alignment between LiDAR points and image pixels and capture the correlation between LiDAR and image dynamic features. These transformer networks have shown promising performance on 3D object detection tasks. However, these networks are large and contain many parameters, so further investigation is needed to develop more lightweight and robust models.

III. METHOD

The proposed model consists of ResNet-50, ResNet-101, and ResNet-152 [8] backbone networks for image feature extraction and the VoxelNet [20] network for the processing of point clouds of LiDAR, as shown in Fig. 1. The proposed multiresolution attention model and DWT and IDWT are integrated with the image backbone to enhance feature extraction capability. The image and the LiDAR point cloud features are voxelized into voxel representation before feeding into voxel encoders. Finally, the transformer decoder, consisting of a multi-head attention network, helps for global interaction between object queries and encoder features and local interaction between generated object queries for accurate and robust prediction.

A. INPUT TRANSFORMATION TO COMMON REPRESENTATION

Recently, several transformer-based 2D object detection models have been proposed following the introduction of DETR [38]. These networks typically use a CNN backbone to extract image features and a transformer network to convert

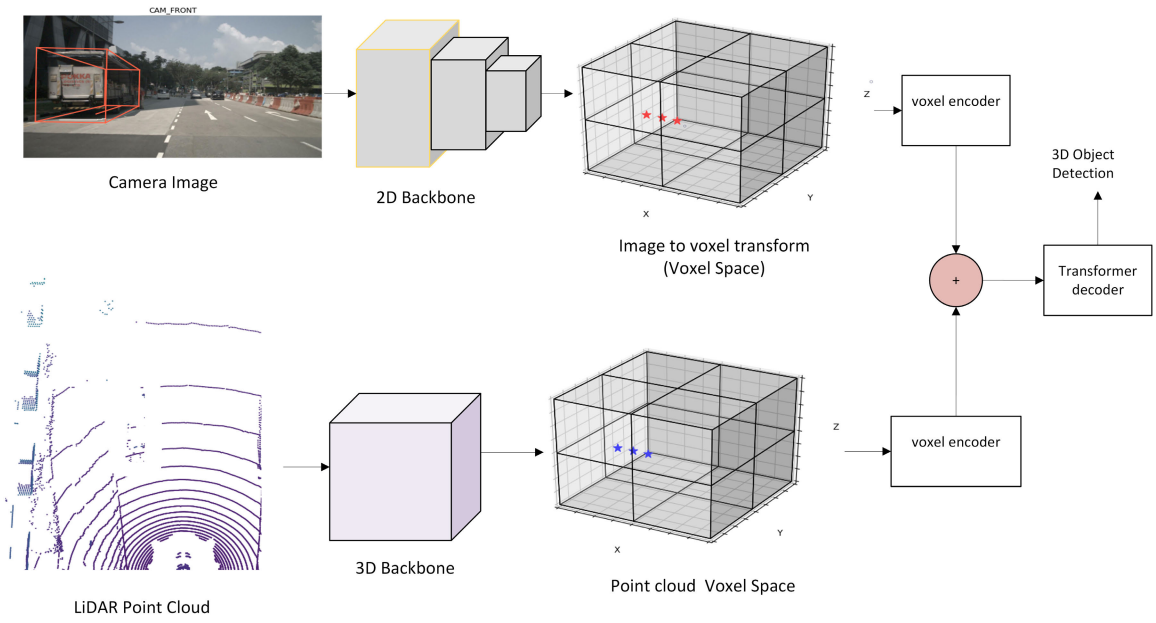

FIGURE 1. The proposed architecture. The CNN backbone network (ResNet-50, ResNet-101, and ResNet-152 [8]) with the proposed attention module extracts the image features before building the voxel features. VoxelNet [20] is a backbone for extracting LiDAR point cloud features. The deformable DETR [42] transformer decoder is also used as a transformer decoder.

the learned embeddings called object queries and their positional information into predictions. However, research on transformer-based 3D object detection has been limited until recently. This work uses the ResNet [8] backbone network with FPN [39] and the proposed multiresolution attention module to extract multiscale image features. The voxel image features are then constructed from these image features by first generating the depth information of each image like in [40] and [41]. The depth information $D_I \in R^{D\times H\times W}$ can be generated using:

$$D_I(u, v) = Softmax\left(Conv(F_I)(u, v)\right), \tag{1}$$

The terms H, W, D, and F_I represent the height, width, depth bins, and image features, respectively. The coordinates in the image plane are indicated by (u, v). D is fixed at 64, following the approach in [6]. The depth distribution corresponding to the image pixel feature can be predicted from D_I. Then, the voxel space features V_I can be generated by

$$V_I(x, y, z) = D_I(u, v, d) \times F_I(u, v), \tag{2}$$

where the sampling point (u, v, d) in the image plane is calculated from the sampling point (x, y, z) with the calibration matrix. d is the depth of reference along the axis D. $D_I(u, v, d)$ is the probability feature occupancy of $F_I(u, v)$ in voxel (x, y, z). Then, multiple voxel spaces are concatenated and features are integrated along the temporal dimension to form a unified voxel space V_I.

For the LiDAR point cloud voxel space, the point cloud is split into several regular voxels. The VoxelNet backbone [20], [21], [22] is used to extract and process input voxels. Multiscale features are generated using parallel heads with various strides like [6]. These multiscale features are upsampled from the voxel space $V_p \in R^{X\times Y\times Z\times C}$. Where X, Y, and Z are coordinates of a voxel in the three-dimensional space. After constructing the voxel spaces for image and LiDAR point cloud data, the voxel encoder, comprising three convolutional blocks, facilitates local feature interaction within each voxel space. Once voxel features are generated, they lack interaction among themselves. Thus, the voxel encoder is crucial in fostering local feature interactions. Then, the processed features are fused to better utilize features from the two voxel spaces. After the fusion stage, the transformer decoder performs object-level interactions for accurate and robust predictions.

B. TRANSFORMER DECODER AND FEED-FORWARD NETWORK

Using transformer networks for machine translation was first introduced by Vaswani et al. [43]. The multi-head attention module helps identify important information within a query element and a set of key elements. This is accomplished by assigning attention weights to query-key pairs. The resulting outputs from various attention heads are combined using learnable weights, encouraging the model to pay attention to distinct representations and positions. If a query element $q \in \Omega_q$ has representation features $z_q \in \mathbb{R}^C$ and $k \in \Omega_k$ indexes with $x_k \in \mathbb{R}^C$, the multi-head attention feature can be computed as follows [42]:

$$MultiHeadAttn(zq, x) = \sum_{m=1}^{M} W_m \left[\sum_{k\in\Omega_k} A_{mqk}.W'_m x_k\right], \tag{3}$$

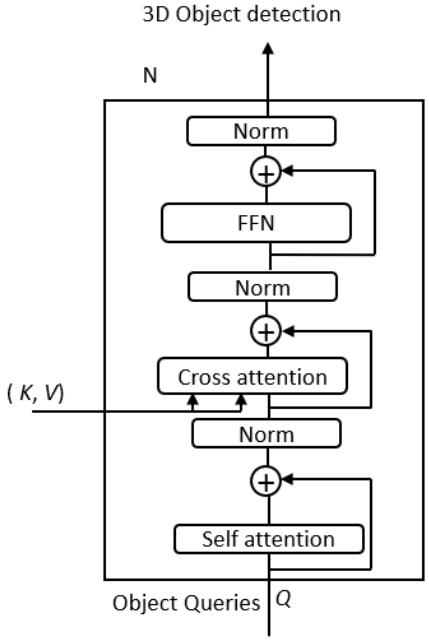

FIGURE 2. Transformer Decoder Network. The self-attention and cross-attention ensure local and global interaction of features to learn the most important features. where *Q*, *K*, and *V* denote query, key, and value.

where C, Ω_q, and Ω_k represent the feature dimension, set of queries, and key elements, respectively. Furthermore, m indexes the attention head and $W'_m \in R^{C_v \times C}$ and $W_m \in R^{C \times C_v}$ are learnable weights with $C_v = C/M$. The attention weights $A_{mqk} \propto exp\left(\frac{z_q^T u_m^T v_m x_k}{\sqrt{c_v}}\right)$, which are normalized. U_m,$V_m \in R^{C_v \times C}$ are also learnable weights.

The DETR transformer decoder is slow and struggles to detect small objects. This issue is solved using the deformable DETR [42]. Deformable DETR and DETR eliminate post-processing steps, such as nonmaximum suppression (NMS), because the Hungarian loss [44] removes duplicates in bipartite matching. The transformer decoder employs d-dimensional positional encoding to embed object queries to obtain an accurate object prediction from the input of object queries. Initially, N object queries $Q \in \mathrm{R}^{N \times C}$ are initialized. Then, self-attention between object queries generates different object candidates. The self-attention module is crucial for local feature interaction between object queries. Before the generation of object queries, residual connection and normalization operations are performed, and fused feature maps are generated through cross-attention to produce more relevant object candidates, as shown in Fig. 2. To achieve global feature interaction, the cross-attention module allows for the interaction of encoder and decoder features rather than just focusing on local features such as self-attention. The combination of self-attention and cross-attention ensures both local and global interaction of features, which is crucial for detecting and localizing the most important features. Residual connection and normalization operations are performed after the cross-attention module, and element-wise summation is used to reduce computation costs for all residual connections. The generated object candidates are fed into the feed-forward network (FFN), a simple multi-layer perceptron (MLP). Finally, the FFN decodes the object queries into predictions for bounding-box information and classification. To refine 3D bounding boxes, iterative box refinement [42] is used. This process ultimately results in accurate and robust object classification with corresponding localization of objects.

C. MULTIRESOLUTION ATTENTION AND BACKBONE NETWORK

Attention mechanisms have been extensively studied [45], [46], [47] to enhance the feature extraction capabilities of CNNs and improve feature representation by suppressing irrelevant features. This work introduces a multiresolution wavelet-based attention module, as shown in Fig. 3 (b). Using the conventional DWT, the approach decomposes an image's frequency components into high- and low-frequency elements at each stage of decomposition. As shown in Fig. 3 (a), the low-frequency component coefficient is decomposed into low- and high-frequency coefficients. The next step is to downsample along the rows and columns by a factor of two, resulting in a total downsampling by a factor of two at each stage. Depending on the level of decomposition, these steps are repeated. For a given image X, the low-frequency component is divided into four frequency subbands at each stage: X_{LL}, X_{LH}, X_{HL}, and X_{HH} as shown in Fig. 3 (a). The X_{LL} (approximate component) represents subbands of the low-frequency component, whereas X_{LH} (horizontal component), X_{HL} (vertical component), and X_{HH} (diagonal component) are that of the high-frequency component. Furthermore, the wavelet transform can separate noise from the image and concentrate in the high-frequency component [7], mainly in X_{HH}. Therefore, X_{HH} is discarded during the operation.

In standard CNNs, high-frequency components are disregarded. However, implementing the DWT makes extracting additional features from the high-frequency subbands X_{LH} and X_{HL} possible, thereby enhancing feature representation. These components, X_{LH} and X_{HL}, are concatenated to enhance features before global average pooling is applied, as shown in Fig. 3 (b). Subsequently, a 1×1 convolution followed by LeakyReLU and another 1×1 convolution is applied before passing through the softmax activation function. Wavelets can have negative coefficients, making LeakyReLU a more suitable choice for activation. The features extracted in this process are multiplied with X_{LL} to create an activation weight. This activation weight is combined with X_{LL} through a skip connection for enriching features. The original features are then reconstructed and preserved by applying an IDWT. This property helps the DWT perform invertible downsampling without losing information.

This work integrates a multiresolution attention module into the ResNet [8] backbone network, incorporating it into both the BasicBlock and Bottleneck blocks of the ResNet

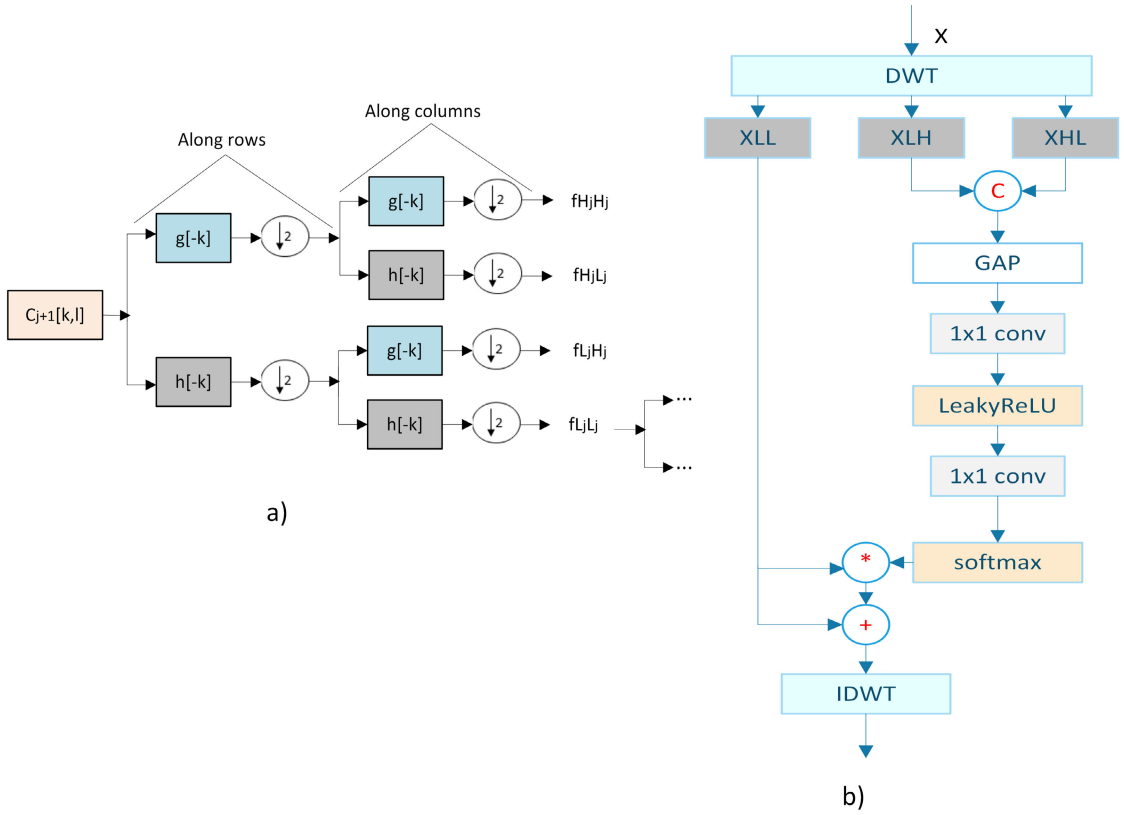

FIGURE 3. a) single-stage 2D wavelet decomposition. The low-frequency component coefficient (*C*) is decomposed into low- and high-frequency coefficients. The next step is to downsample along the rows and columns by a factor of two, totaling two downsamplings. Depending on the level of decomposition, these steps are repeated. *g* and *h* denote the high-frequency filter and the low-frequency filter. b) Multiresolution attention module. GAP and C represent global average pooling and concatenation. * and + indicate element-wise multiplication and element-wise addition, respectively.

architecture. The attention module is designed to mirror the repetitive structure of blocks in the ResNet network. This module leverages the inherent properties of wavelets, such as multiresolution, invertible downsampling, noise separation capability, and sparsity, to enrich features while mitigating noise effects. The invertible downsampling property of wavelets facilitates the downsampling of features without information loss, as these features can be restored using IDWT. This approach addresses the issue of information loss in standard CNN models, which often complicates classification and detection tasks, particularly for small objects. Furthermore, extracting features from high-frequency components provides additional information typically overlooked in standard CNN networks. The low-frequency component, similar to the features used in CNNs, means that high-frequency-derived features act as supplementary enhancements to the overall feature set, thus improving performance. Therefore, an attention mechanism built upon these wavelet properties not only boosts the performance of convolutional networks but does so without incurring additional computational costs compared to CNN-based attention modules. The input and output dimensions of the attention module remain consistent. Therefore, it can be seamlessly integrated into any CNN-based backbone network.

Additionally, the DWT and IDWT are integrated into the ResNet backbone network's BasicBlock and Bottleneck blocks, enhancing its feature extraction capabilities, as shown in Fig. 4. The BasicBlock, a key component of ResNet, includes two layers of 3×3 convolutions, with each layer followed by batch normalization and a ReLU activation function. In contrast, the Bottleneck block, another crucial element of ResNet, comprises two 1 × 1 convolution layers and a central 3 × 3 convolution layer, each accompanied by batch normalization and ReLU activation. The DWT is applied as an invertible downsampling block, whereas the IDWT is utilized at the end of both block types to restore and maintain detail. This simple incorporation showcases the potential of DWT and IDWT to seamlessly integrate with existing CNN architectures, thus significantly enhancing their feature extraction capacity. The results of these experimental integrations are detailed in Section IV.

D. NETWORK OPTIMIZATION

Network optimization techniques, such as pruning, quantization, and quantization-aware training (QAT), were employed to enhance the proposed model's efficiency and performance. Pruning involves removing less important neurons or connections from a network, which minimizes its complexity.

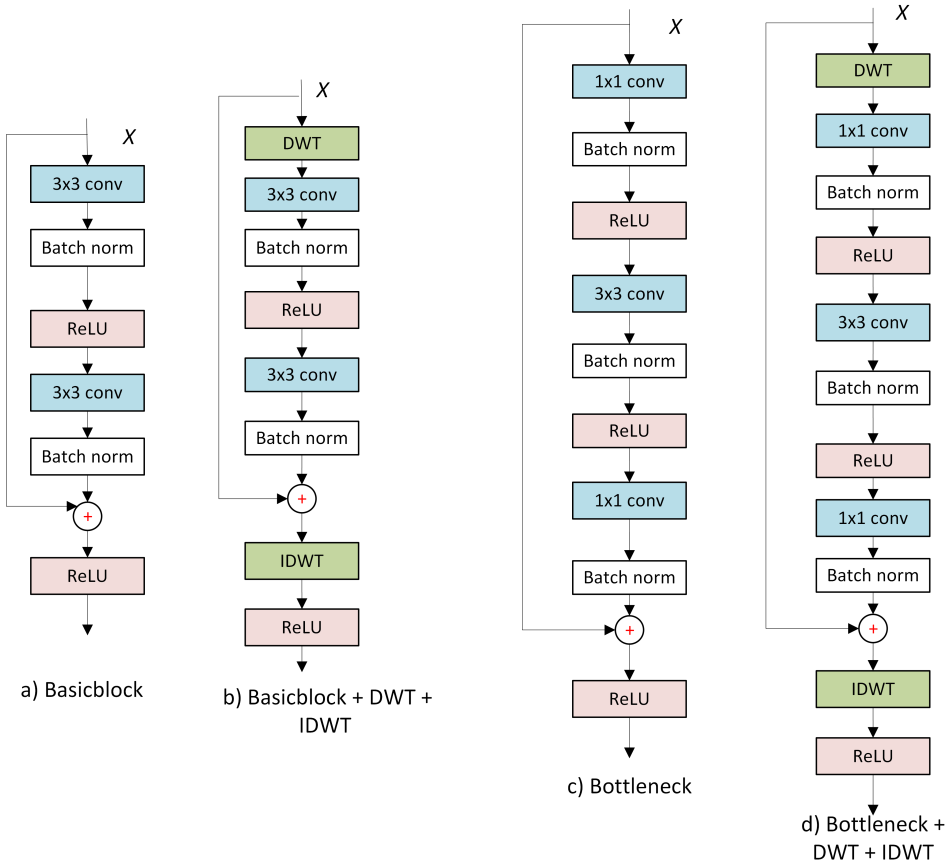

FIGURE 4. ResNet Backbone Network's BasicBlock and Bottleneck with DWT and IDWT integration.

By simplifying the model, this method lowers processing requirements without sacrificing significant accuracy. Pruning can be groped into unstructured or structured [48]. Structured pruning removes individual units or structures—like neurons, channels, or layers—from a model based on how much they contribute to its functionality. On the other hand, unstructured pruning focuses on removing individual weights and frequently results in a more effective model architecture with less computing complexity. This work uses unstructured pruning, which removes individual weights from the model by determining the least important weights.

Quantization reduces the memory and computational requirements of the model by converting floating-point weights and activations into lower-bit representations, like 8-bit integer representations. Although static quantization reduces computational complexity, it leads to a performance decline. Therefore, QAT is implemented in the proposed model to enhance its performance further by mimicking the effect of quantization during training to simulate the actual quantization so that the model will have less performance drop during inference. This method successfully avoids significant performance deterioration after quantization by ensuring the model is optimized for quantization. It can, therefore, adapt to decreased precision with good performance. The Straight Through Estimator (STE) [49] is used in QAT to avoid the zero gradient issue. This optimization approach balances high-level data processing and computing efficiency, which enhances the model's deployability and functionality.

IV. RESULTS AND DISCUSSIONS

A. DATASET AND IMPLEMENTATION DETAILS

The proposed work is trained and evaluated using the nuScenes [50] dataset, a large-scale autonomous driving dataset of 700, 150, and 150 scenes in the training, validation, and test set, respectively. There are ten different object classes in the dataset. The main evaluation metrics for the dataset's 3D object detection category are the mean average precision (mAP) and nuScenes detection score (NDS). To separate the effects of orientation and object size for detection, the AP approach of the nuScenes dataset specifies a match between a prediction and ground truth by thresholding the 2-D center distance d on the ground plane instead of IOU [1], [50].

$$mAP = \frac{1}{|C||D|} \sum_{c \in C} \sum_{d \in D} AP_{c,d}, \tag{4}$$

where $D = \{0.5, 1, 2, 4\}$ meters, and C is the set of classes.

Similarly, the nuScenes detection score (NDS) is computed.

$$NDS = \frac{1}{10}\left[5mAP + \sum_{mTP \in TP} (1 - min(1, mTP))\right], \tag{5}$$

TABLE 1. The nuScenes test set benchmark was employed to evaluate various methods. V0.075 denotes a voxel grid size of 0.075 meters, used in conjunction with the VoxelNet backbone. Here, 'L' and 'C' represent LiDAR point cloud and camera data, and 'LC' denotes both. The best values are highlighted in blue, and the second-best in bold. DLA34 [52], V2-99 [53], and ResNext-101 [54] are Backbone networks.

Method	Backbone	Modality	NDS (%)	mAP (%)
BEVDet [55]	V2-99	C	48.2	42.2
M^2BEV [56]	ResNeXt-101	C	47.4	42.9
BEVFormer [57]	V2-99	C	53.5	44.5
BEVDet4D [58]	V2-99	C	56.9	45.1
UVTR-L2C [6]	V2-99	C	52.2	45.2
MFusion [36]	V2-99	C	52.8	**45.3**
Ours*	V2-99	C	**53.7**	45.5
CenterPoint [59]	V0.075	L	67.3	60.3
UVTR-M [6]	V0.075	L	67.6	60.8
MVP [60]	V0.075	L	67.6	60.8
MFusion [36]	V0.075	L	67.8	63.4
TransFusion [4]	V0.075	L	70.2	**65.5**
Ours*	V0.075	L	**69.8**	65.7
MVP [60]	V0.075-DLA34	LC	70.5	66.4
PointAugmenting [31]	V0.075-DLA34	LC	71.0	66.8
UVTR-M [6]	V0.075-R101	LC	70.2	65.4
MFusion [36]	V0.075-R101	LC	71.3	67.7
TransFusion [4]	V0.075-DLA34	LC	71.7	68.9
BEVFusion [37]	V0.075-Swin-T	LC	72.9	70.2
Ours*	V0.075-R50	LC	70.7	66.9
Ours*	V0.075-R101	LC	71.8	68.4
Ours*	V0.075-R152	LC	**72.2**	**69.3**

TABLE 2. Ablation studies. MR is the proposed multiresolution attention module.

Backbone	MR	DWT/IDWT	NDS (%)	mAP (%)
V0.075-R50	-	-	70.1	66.2
	✓	-	70.5	66.7
	-	✓	70.4	66.6
	✓	✓	70.7	66.9
V0.075-R101	-	-	71.3	67.7
	✓	-	71.5	68.1
	-	✓	71.5	68.0
	✓	✓	71.8	68.4
V0.075-R152	-	-	71.5	68.1
	✓	-	72.0	69.0
	-	✓	71.7	68.8
	✓	✓	72.2	69.3

TABLE 3. Pruning of the proposed multimodal fusion model.

Backbone	Pruning Ratio(%)	NDS (%)	mAP (%)
V0.075-R50	-	70.6	66.9
	10	70.5	66.9
	50	65.6	62.4
	80	56.3	52.5
	90	43.6	37.8
V0.075-R101	-	71.7	68.1
	10	71.7	68.1
	50	66.7	64.0
	80	59.2	55.3
	90	47.8	42.5

TABLE 4. Quantization and Quantization Aware Training.

Backbone	Quantization	NDS (%)	mAP (%)
V0.075-R50	quantization	59.8	56.1
	QAT	67.6	64.2

where the mean True positive (mTP) can be expressed as:

$$mTP = \frac{1}{|C|}\sum_{c\in C} TP_c, \tag{6}$$

Point cloud ranges of $[-54, 54]$, $[-54, 54]$, and $[-5, 3]$ meters for X-, Y-, and Z-axis, respectively, are used. The voxel size is set to (0.075m, 0.075m, 0.2m), which is commonly used. Additionally, 900 object queries were used for the experiment. The AdamW optimization with an initial learning rate of $1e^{-5}$ and a gamma of 0.1 is used for the experiments. For LiDAR data, sweeps of 10 are used. The model was trained for 36 epochs using NVidia A100/80G GPUs running on CentOs 8. The work is implemented using MMDetection3D [51] open-source toolbox.

B. QUANTITATIVE ANALYSIS

The advantages of multimodal fusion go beyond performance enhancement. They are especially essential in robotics applications like real-time autonomous driving, where the possibility of sensor failure exists. In such instances, the functionality of the remaining sensors becomes crucial, even when the system lacks complete data for decision-making. Multimodal fusion improves performance by integrating diverse sensor data and ensures system reliability. This emphasizes the necessity of equipping autonomous vehicles with multiple sensors for practical applications. The comparison of various methods is presented in Table 1. Techniques such as Transfusion [4] and UVTR [6] incorporate transformers, while others rely on convolutional neural networks without requiring transformers. Additional methods like BEVFusion [37] employ an end-to-end transformer for feature extraction and as a transformer head. Evaluation through NDS and mAP metrics shows that multimodal fusion surpasses the performance of methods using only LiDAR or cameras.

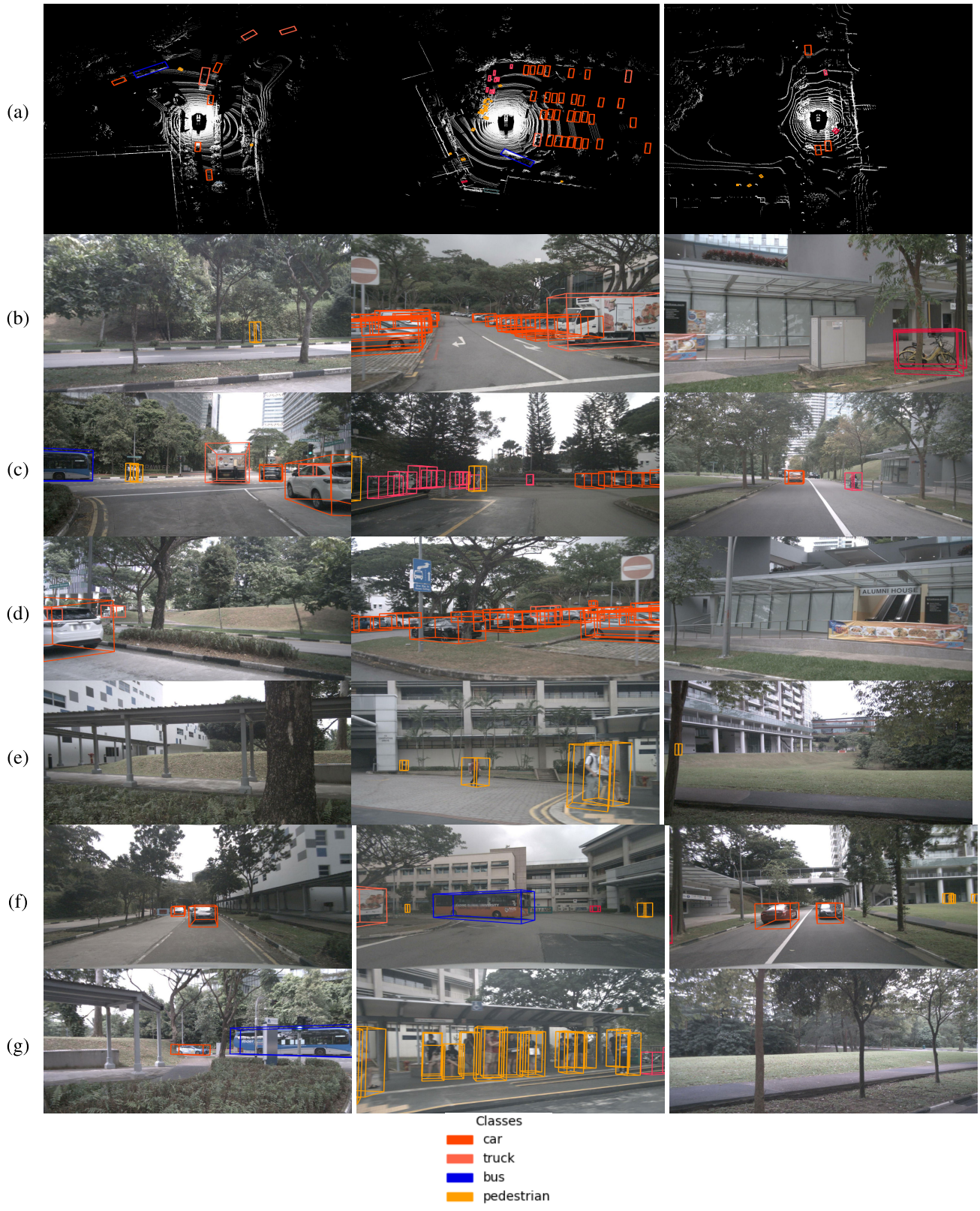

FIGURE 5. Qualitative output of 3D object detection on the nuScenes dataset. (a) BEV multimodality, (b) front left camera, (c) front camera, (d) front right camera, (e) back left camera, (f) back camera and (g) back right camera.

Table 1 demonstrates the proposed model's results trained with various backbone networks. Backbones such as ResNet-50, ResNet-101, and ResNet-152, paired with the VoxelNet backbone, delivered performance comparable to state-of-the-art models. The LiDAR-only approach yielded 69.8% in NDS and 65.7% in mAP. In contrast, the

camera-only method achieved 53.7% in NDS and 45.5% in mAP. Additionally, multimodal fusion employing the ResNet-50 backbone for images and VoxelNet for point clouds resulted in 70.6% for NDS and 66.9% for mAP. This performance improved to 71.8% in NDS and 68.4% in mAP with the ResNet-101 backbone. The highest enhancement was observed with the ResNet-152 backbone, reaching 72.2% in NDS and 69.3% in mAP. The multimodal fusion approach demonstrated a notable performance improvement compared to the LiDAR-only method, with increases of up to 2.4% in NDS and 3.6% in mAP. Furthermore, it showed a substantial enhancement of 18.5% in NDS and 23.8% in mAP over the camera-only methods. It is worth mentioning that BEVFusion outperforms our methods in multimodal fusion, although it is an end-to-end transformed model that is more computationally intensive.

C. ABLATION STUDIES AND NETWORK OPTIMIZATION

The model was trained both with and without the proposed multiresolution attention module, along with the integration of DWT/IDWT into the backbone network, to showcase the effectiveness of these solutions. As presented in Table 2, the NDS and mAP accuracies indicate improved model performance upon integrating the attention module and DWT/IDWT into the backbone network.

Both proposed solutions demonstrated improved performance when tested independently. Using the V0.075-R50 backbone, the multimodal fusion method exhibited a 0.6% increase in NDS and a 0.7% rise in mAP. On the other hand, with the V0.075-R152 backbone, it achieved a 0.7% rise in NDS and 1.2% in mAP. These outcomes prove the effectiveness of the proposed attention module and the integration of DWT/IDWT in improving the model's feature extraction capabilities. Wavelets can capture coarse and fine details, and the inherent multiresolution analysis seamlessly integrates with the neural network's capacity to extract features across various scales. This enables the neural network to focus on essential features, resulting in an efficient and effective feature extraction. Moreover, the invertible downsampling nature of wavelets plays an essential role in preserving information during downsampling. Consequently, the multiresolution attention module and DWT/IDWT facilitate invertible downsampling and reduce computations while preserving crucial information, which is unachievable with standard CNNs.

Despite the performance enhancements of multimodal 3D fusion models, their high computational complexity and high number of parameters often make them impractical for real-time implementation. This issue worsens with transformer networks due to the computationally intensive multihead self-attention. This work tackles these challenges by employing network optimization techniques, including pruning and quantization, using the native PyTorch API.

Table 3 presents the results of running the model with varying pruning ratios, utilizing ResNet-50 and ResNet-101 as image backbones and VoxelNet as the LiDAR backbone for multimodal fusion. Pruning effectively reduces the model's computational load and memory requirements. However, a notable performance decline is observed as the pruning ratio increases. The accuracy remains nearly comparable to the unpruned model at a 10% pruning ratio, but it deteriorates further with higher proportions of 50%, 80%, and 90%. This performance drop is particularly significant at a 90% pruning ratio.

Similarly, the model quantization reduces computational load and memory requirements by employing a lower-bit representation. The floating-point representations are converted into INT8, a native PyTorch integer representation. The ResNet-50 image backbone combined with the VoxelNet LiDAR backbone model is selected for its simplicity. The memory requirement can be reduced four times by converting the 32-bit floating-point representation into an 8-bit integer representation. However, this conversion decreases the model's performance, resulting in 59.8% for NDS and 56.1% for mAP, as shown in Table 4. Finally, QAT is employed to alleviate the accuracy decline caused by quantization. The model was retrained for 12 epochs to adjust for the lost information dynamically. As shown in Table 4, the outcomes demonstrate a significant improvement in accuracy using QAT, enhancing the model's performance by up to 7.8% in NDS and 8.1% in mAP. Therefore, QAT can optimize model performance with less memory and computational resources.

D. VISUALIZATION

The qualitative results presented in Fig. 5 are organized into three columns, showing various prediction outcomes on BEV and the corresponding individual camera views. This figure illustrates multimodal predictions in BEV and each associated camera perspective: Front Left, Front, Front Right, Back Left, Back, and Back Right. The model effectively detects distant objects and accurately identifies objects in crowded scenes.

V. CONCLUSION AND FUTURE WORK

The proposed multimodal fusion 3D object detection model demonstrates promising results on the nuScenes dataset. It transforms camera images and LiDAR point cloud data into a voxel-based representation, then processed by encoder networks. This processing reduces semantic ambiguity resulting from variations in the input data and improves spatial interaction. In the transformer decoder network, object-level interaction, facilitated by self-attention and cross-attention mechanisms, significantly boosts detection capabilities. Cross-attention, in particular, fosters global interaction between the identified objects and the encoder features, leading to robust and accurate detection. Furthermore, the introduced multiresolution attention module and integration of DWT/IDWT with the backbone network improve feature extraction. Network optimization techniques like pruning and QAT substantially reduce memory requirements and computational resources. These techniques are crucial for maintaining competitive performance, especially

if implemented during training to compensate for information loss. Performance evaluations using NDS and mAP metrics on the nuScenes dataset indicate that our model achieves accuracy comparable to state-of-the-art multimodal fusion 3D object detection methods.

The potential for sharing lower-level feature representations across multiple tasks has been established in numerous studies. Multitask and multimodal fusion could enhance performance and optimize resource utilization rather than separately addressing each task, such as 3D detection and segmentation. Furthermore, knowledge distillation among different inputs, such as LiDAR point clouds and camera images, could further improve performance by leveraging the complementary strengths of each modality. These aspects present promising directions for future research.

REFERENCES

[1] S. Y. Alaba and J. E. Ball, "Deep learning-based image 3-D object detection for autonomous driving: Review," *IEEE Sensors J.*, vol. 23, no. 4, pp. 3378–3394, Feb. 2023.

[2] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3D object detection network for autonomous driving," in *Proc. CVPR*, Jul. 2017, pp. 1907–1915.

[3] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, "Joint 3D proposal generation and object detection from view aggregation," *IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, pp. 1–8, Oct. 2018.

[4] 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," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 1080–1089.

[5] B. Xu and Z. Chen, "Multi-level fusion based 3D object detection from monocular images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2345–2353.

[6] Y. Li, Y. Chen, X. Qi, Z. Li, J. Sun, and J. Jia, "Unifying voxel-based representation with transformer for 3D object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 18442–18455.

[7] C. S. Burrus, *Wavelets and Wavelet Transforms*. Houston, TX, USA: Rice Univ. Press, 2015.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[9] T. Liang, J. Glossner, L. Wang, S. Shi, and X. Zhang, "Pruning and quantization for deep neural network acceleration: A survey," *Neurocomputing*, vol. 461, pp. 370–403, Oct. 2021.

[10] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell, and K. Q. Weinberger, "Pseudo-LiDAR from visual depth estimation: Bridging the gap in 3D object detection for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8437–8445.

[11] X. Ma, Z. Wang, H. Li, P. Zhang, W. Ouyang, and X. Fan, "Accurate monocular 3D object detection via color-embedded 3D reconstruction for autonomous driving," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6850–6859.

[12] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, "3D object proposals for accurate object class detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 424–432.

[13] Y. Shi, Y. Guo, Z. Mi, and X. Li, "Stereo CenterNet based 3D object detection for autonomous driving," 2021, *arXiv:2103.11071*.

[14] W. Peng, H. Pan, H. Liu, and Y. Sun, "IDA-3D: Instance-depth-aware 3D object detection from stereo vision for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13012–13021.

[15] T. He and S. Soatto, "Mono3D++: Monocular 3D vehicle detection with two-scale 3D hypotheses and task priors," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 8409–8416.

[16] H. N. Hu, Q. Z. Cai, D. Wang, J. Lin, M. Sun, P. Krahenbuhl, T. Darrell, and F. Yu, "Joint monocular 3D vehicle detection and tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Seoul, South Korea, Oct./Nov. 2019, pp. 5390–5399.

[17] P. Li, H. Zhao, P. Liu, and F. Cao, "RTM3D: Real-time monocular 3D detection from object keypoints for autonomous driving," 2020, *arXiv:2001.03343*.

[18] S. Y. Alaba and J. E. Ball, "A survey on deep-learning-based LiDAR 3D object detection for autonomous driving," *Sensors*, vol. 22, no. 24, p. 9577, Dec. 2022.

[19] A. Barrera, C. Guindel, J. Beltrán, and F. García, "BirdNet+: End-to-end 3D object detection in LiDAR bird's eye view," in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst. (ITSC)*, Sep. 2020, pp. 1–6.

[20] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4490–4499.

[21] Y. Yan, Y. Mao, and B. Li, "SECOND: Sparsely embedded convolutional detection," *Sensors*, vol. 18, no. 10, p. 3337, Oct. 2018.

[22] B. Graham and L. van der Maaten, "Submanifold sparse convolutional networks," 2017, *arXiv:1706.01307*.

[23] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 12697–12705.

[24] S. Y. Alaba and J. E. Ball, "WCNN3D: Wavelet convolutional neural network-based 3D object detection for autonomous driving," *Sensors*, vol. 22, no. 18, p. 7010, Sep. 2022.

[25] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 77–85.

[26] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," 2017, *arXiv:1706.02413*.

[27] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "PV-RCNN: Point-voxel feature set abstraction for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10526–10535.

[28] S. Y. Alaba, A. Gurbuz, and J. E. Ball, "A comprehensive survey of deep learning multisensor fusion-based 3D object detection for autonomous driving: Methods, challenges, open issues, and future directions," *TechRxiv*, Aug. 2022.

[29] C.-H. Wang, H.-W. Chen, and L.-C. Fu, "VPFNet: Voxel-pixel fusion network for multi-class 3D object detection," 2021, *arXiv:2111.00966*.

[30] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "PointPainting: Sequential fusion for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4603–4611.

[31] C. Wang, C. Ma, M. Zhu, and X. Yang, "PointAugmenting: Cross-modal augmentation for 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11789–11798.

[32] Y. Wang, X. Chen, L. Cao, W. Huang, F. Sun, and Y. Wang, "Multimodal token fusion for vision transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 12176–12185.

[33] Y. Wang, T. Ye, L. Cao, W. Huang, F. Sun, F. He, and D. Tao, "Bridged transformer for vision and point cloud 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 12104–12113.

[34] M. Liu, J. Ma, Q. Zheng, Y. Liu, and G. Shi, "3D object detection based on attention and multi-scale feature fusion," *Sensors*, vol. 22, no. 10, p. 3935, May 2022.

[35] Y. Li, A. W. Yu, T. Meng, B. Caine, J. Ngiam, D. Peng, J. Shen, Y. Lu, D. Zhou, Q. V. Le, A. Yuille, and M. Tan, "DeepFusion: LiDAR-camera deep fusion for multi-modal 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17161–17170.

[36] S. Y. Alaba and J. E. Ball, "Multi-sensor fusion 3D object detection for autonomous driving," in *Autonomous Systems: Sensors, Processing and Security for Ground, Air, Sea and Space Vehicles and Infrastructure*, vol. 12540. Bellingham, WA, USA: SPIE, 2023, pp. 36–43.

[37] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. L. Rus, and S. Han, "BEVFusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 2774–2781.

[38] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K. Cham, Switzerland: Springer, Aug. 2020, pp. 213–229.

[39] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.
[40] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3D," in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV)*, Glasgow, U.K., Aug. 2020, pp. 194–210.
[41] C. Reading, A. Harakeh, J. Chae, and S. L. Waslander, "Categorical depth distribution network for monocular 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 8551–8560.
[42] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," 2020, *arXiv:2010.04159*.
[43] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.
[44] H. W. Kuhn, "The Hungarian method for the assignment problem," *Nav. Res. Logistics Quart.*, vol. 52, no. 1, pp. 7–21, Feb. 2005.
[45] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 3–19.
[46] X. Zhao, P. Huang, and X. Shu, "Wavelet-attention CNN for image classification," *Multimedia Syst.*, vol. 28, no. 3, pp. 915–924, Jun. 2022.
[47] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
[48] P. Ganesh, Y. Chen, X. Lou, M. A. Khan, Y. Yang, H. Sajjad, P. Nakov, D. Chen, and M. Winslett, "Compressing large-scale transformer-based models: A case study on BERT," *Trans. Assoc. Comput. Linguistics*, vol. 9, pp. 1061–1080, Sep. 2021.
[49] Y. Bengio, N. Léonard, and A. Courville, "Estimating or propagating gradients through stochastic neurons for conditional computation," 2013, *arXiv:1308.3432*.
[50] 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," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11618–11628.
[51] MMDetection3D Contributors. (2020). *MMDetection3D: OpenMMLab Next-Generation Platform for General 3D Object Detection*. [Online]. Available: https://github.com/open-mmlab/mmdetection3d
[52] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2403–2412.
[53] Y. Lee, J.-W. Hwang, S. Lee, Y. Bae, and J. Park, "An energy and GPU-computation efficient backbone network for real-time object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 752–760.
[54] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5987–5995.
[55] J. Huang, G. Huang, Z. Zhu, Y. Ye, and D. Du, "BEVDet: High-performance multi-camera 3D object detection in bird-eye-view," 2021, *arXiv:2112.11790*.
[56] E. Xie, Z. Yu, D. Zhou, J. Philion, A. Anandkumar, S. Fidler, P. Luo, and J. M. Alvarez, "M^2BEV: Multi-camera joint 3D detection and segmentation with unified birds-eye view representation," 2022, *arXiv:2204.05088*.
[57] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Q. Yu, and J. Dai, "BEVFormer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," 2022, *arXiv:2203.17270*.
[58] J. Huang and G. Huang, "BEVDet4D: Exploit temporal cues in multi-camera 3D object detection," 2022, *arXiv:2203.17054*.
[59] T. Yin, X. Zhou, and P. Krähenbühl, "Center-based 3D object detection and tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11779–11788.
[60] T. Yin, X. Zhou, and P. Krähenbühl, "Multimodal virtual point 3D detection," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 16494–16507.

SIMEGNEW YIHUNIE ALABA (Member, IEEE) received the Bachelor of Science degree in electrical engineering from Arba Minch University and the Master of Science degree in computer engineering from Addis Ababa University, Ethiopia. He is currently pursuing the Ph.D. degree in electrical and computer engineering with Mississippi State University. His academic focus encompasses machine learning, deep learning, computer vision, and autonomous driving.

JOHN E. BALL (Senior Member, IEEE) received the B.S. and Ph.D. degrees in electrical engineering from Mississippi State University, in 1991 and 2007, respectively, and the M.S. degree in electrical engineering from Georgia Institute of Technology, in 1993. Currently, he is an Associate Professor and the Endowed Chair of Electrical and Computer Engineering with Mississippi State University. His research interests include sensors, sensor processing, deep learning, and autonomous vehicles, especially in an unstructured environment. He serves as an Associate Editor for IEEE Signal Processing Letters and the *Journal of Applied Remote Sensing*.