World Electric Vehicle Journal

Article

Att-BEVFusion: An Object Detection Algorithm for Camera and LiDAR Fusion Under BEV Features

Peicheng Shi [1,*], Mengru Zhou [1], Xinlong Dong [1] and Aixi Yang [2]

[1] School of Mechanical and Automotive Engineering, Anhui Polytechnic University, Wuhu 241000, China; dream3219271439@163.com (M.Z.); dxl007ha@163.com (X.D.)
[2] Polytechnic Institute, Zhejiang University, Hangzhou 310015, China; yangaixi@zju.edu.cn
[*] Correspondence: shipeicheng@126.com or shipeicheng@ahpu.edu.cn

Abstract: To improve the accuracy of detecting small and long-distance objects while self-driving cars are in motion, in this paper, we propose a 3D object detection method, Att-BEVFusion, which fuses camera and LiDAR data in a bird's-eye view (BEV). First, the transformation from the camera view to the BEV space is achieved through an implicit supervision-based method, and then the LiDAR BEV feature point cloud is voxelized and converted into BEV features. Then, a channel attention mechanism is introduced to design a BEV feature fusion network to realize the fusion of camera BEV feature space and LiDAR BEV feature space. Finally, regarding the issue of insufficient global reasoning in the BEV fusion features generated by the channel attention mechanism, as well as the challenge of inadequate interaction between features. We further develop a BEV self-attention mechanism to apply global operations on the features. This paper evaluates the effectiveness of the Att-BEVFusion fusion algorithm on the nuScenes dataset, and the results demonstrate that the algorithm achieved 72.0% mean average precision (mAP) and 74.3% nuScenes detection score (NDS), with an advanced detection accuracy of 88.9% and 91.8% for single-item detection of automotive and pedestrian categories, respectively.

Keywords: autonomous car; BEV feature fusion; object detection

check for updates

Citation: Shi, P.; Zhou, M.; Dong, X.; Yang, A. Att-BEVFusion: An Object Detection Algorithm for Camera and LiDAR Fusion Under BEV Features. *World Electr. Veh. J.* **2024**, *15*, 539. https://doi.org/10.3390/wevj15110539

Academic Editor: Joeri Van Mierlo

Received: 22 October 2024
Revised: 11 November 2024
Accepted: 12 November 2024
Published: 20 November 2024

1. Introduction

Environment sensing as a basis for unmanned systems can understand and adapt to the surrounding environment, drawing increasing attention from both industry and academia. An accurate and comprehensive understanding of the driving environment around a person, including vehicles, pedestrians, and streets, is essential for self-driving cars to make reliable and efficient driving decisions. Among these tasks, 3D object detection is a core aspect of 3D perception. However, due to the inherent properties of different sensors, relying only on a single type of sensor cannot ensure stable and high-quality perception results in the ever-changing driving environment [1]. For example, camera data can provide dense color and texture information but are unable to capture depth; radar can detect objects at long distances and provide real-time object information but has lower resolution, limiting high-precision data; and LiDAR offers accurate depth and structural information but faces limitations in range and sparsity [2]. It is worth noting that multi-sensor fusion is important for accurate and reliable sensing since today's self-driving systems are equipped with various sensors, and different sensors can provide complementary signals.

Currently, most sensor systems for driverless vehicles use two sensors, LiDAR and camera, and the fusion of the two can be classified into pre-, mid-, and post-fusion according to the data fusion stage, and several fusion methods have their own advantages and disadvantages. Pre-fusion methods include PointPainting [3] and PointAugmenting [4], which are used to construct complex mapping relationships, mapping point cloud data to image data, or mapping image data to point cloud data. Traditional mid-term fusion methods at the feature end include MV3D [5] and AVOD [6], which use (Convolutional Neural Network) CNN [7] to extract features from the point cloud and RGB images and input them into RPN for fusion. The late fusion methods on the decision side include CLOCs [8] and Fast CLOCs [9], which use a low-complexity multi-modal fusion structure to take the consistency relationship between independent point cloud detection and image detection candidates and input them into sparse convolutional computation to achieve the final fusion result.

Based on the data structure processed by the algorithms, the methods for 3D object detection can be classified into three groups: the first one is based on the original point cloud, the second one is based on the voxel grid, and the third one is based on the bird's-eye view (BEV). Three-dimensional object detection methods based on raw point clouds include PointNet [10], PointNet++ [11], and Point-RCNN [12]. These methods directly utilize raw point clouds without converting them into other grid representations, maximizing the retention of original geometric details. Voxel grid-based 3D object detection methods include SECOND [13] and PointPillars [14]. Unlike methods that use raw point clouds directly, these approaches address challenges posed by the large, unordered, and uneven distribution of point cloud data by partitioning it along the X, Y, and Z axes, converting it into a grid-like encoded representation for more efficient data processing. BEV 3D object detection methods include BEVDet [15] and Bevpool [16]. These methods convert features from the image view into a BEV perspective and use a prediction head for object detection in the BEV, allowing for a more comprehensive representation of the complete scene and effectively enhancing detection performance.

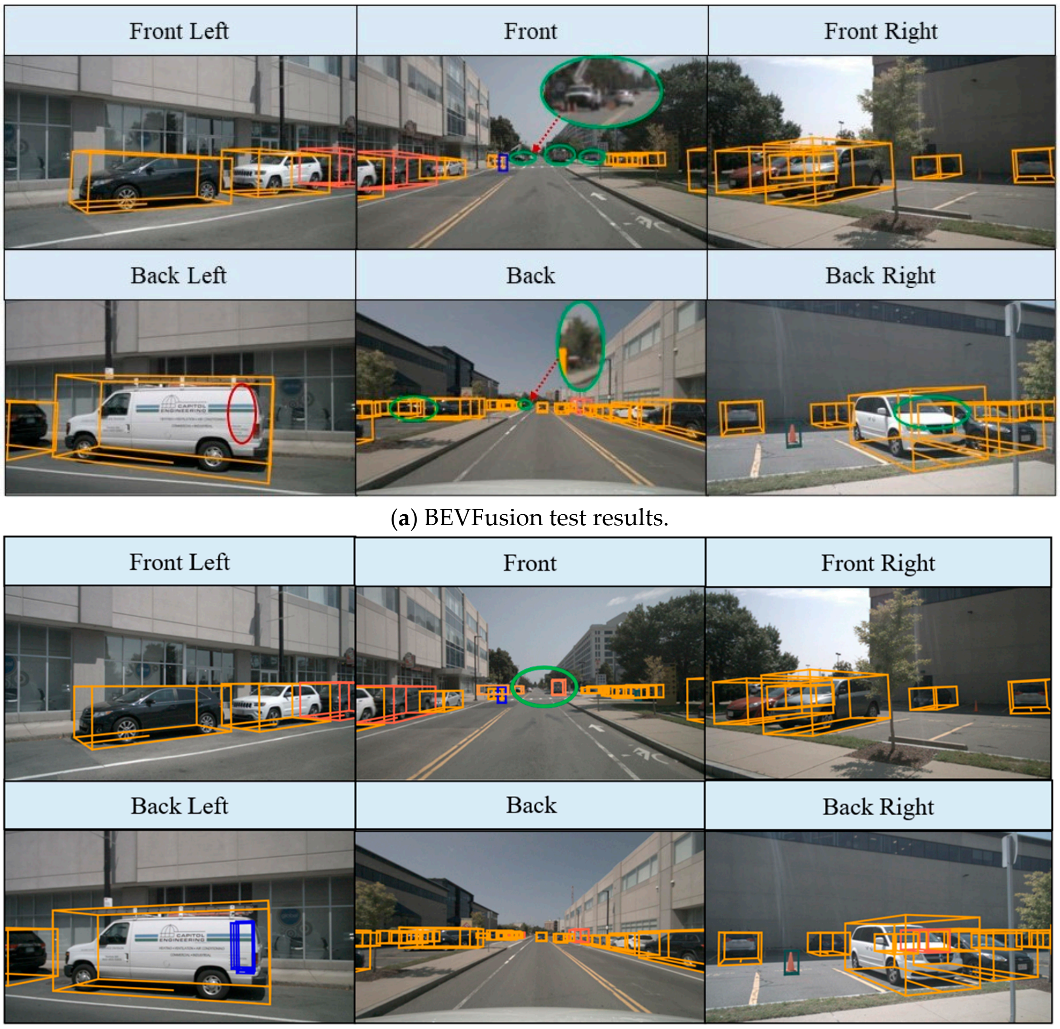
Despite the different implementation methods, the ultimate goal of these approaches is to find an optimal balance between precise features obtained from point clouds and computational cost. While raw point cloud methods maximize the retention of geometric information, they also incur the highest computational overhead. Voxel-based methods reduce complexity and increase computation speed but may result in some loss of information. At this point, the fusion from the BEV perspective plays a significant role; it provides a physically interpretable principle when merging different views, modalities, temporal sequences, and feature information. For example, DeepFusion [17] integrates LiDAR and camera features in a multi-scale space by consolidating them in the BEV, enhancing multi-modal 3D object detection. CBGN [18], also based on the bird's-eye view, performs cross-modal fusion of data from cameras and LiDAR in the BEV space, accurately combining data from two different sensors. However, the low resolution caused by convolution and pooling operations can result in the neglect or loss of small objects, creating challenges in detecting long-distance and small objects due to the limited pixel information.

The most popular work on fusing camera and LiDAR data in the BEV perspective is BEVFusion [19], which utilizes shared image BEV features combined with point cloud features, effectively preserving both geometric and semantic information, achieving excellent results in the nuScenes [20] detection benchmark. However, when merging image BEV features with LiDAR BEV features, spatial alignment errors between the two can lead to the misalignment of features, resulting in lower detection accuracy [21]. As shown in Figure 1, which illustrates the qualitative comparison between BEVFusion and Att-BEVFusion, it can be seen that BEVFusion experiences missed detections for distant and small objects, whereas our proposed Att-BEVFusion incorporates channel attention and self-attention mechanisms, making it more robust in detection accuracy and capable of accurately detecting those missed objects. In this paper, we present an object detection method (Att-BEVFusion) for camera and LiDAR fusion in a BEV perspective, with the main contributions in the following aspects:

(1) An implicit module was developed to convert camera view features into BEV features, along with a module for transforming LiDAR point cloud data into BEV features. This facilitates the alignment of camera view features with LiDAR BEV features, enhancing feature fusion.

(2) An attention mechanism was added. During the construction of the BEV feature fusion module for camera and LiDAR data, a channel attention mechanism was introduced to capture important features. The issue caused by the channel attention mechanism, namely the neglect of global information in the feature map, is addressed by further designing a feature fusion-based self-attention mechanism. This helps to avoid limitations when handling long-range dependencies and the gradual loss of information during transmission.

(3) We trained, validated, and tested our algorithms on the autonomous driving dataset nuScenes. Experiments show that the detection accuracy of the proposed Att-BEVFusion approach outperforms the most popular publicly available results, achieving an outstanding performance of 72.0% mAP and 74.3% NDS in the 3D object detection task, which is of great significance for enhancing the robustness and reliability of intelligent vehicle perception.



(**a**) BEVFusion test results.

(**b**) Att-BEVFusion (Ous) test results.

Figure 1. Comparison between BEVFusion and our proposed method, Att-BEVFusion, which shows that our method is able to effectively detect both distant and occluded objects.

2. Related Work

2.1. Camera-Based 3D Object Detection

With the emergence of Faster-RCNN [22], various methods for 2D object detection have emerged, but the 2D detection results are far from satisfying our needs for unmanned vehicles. In the real 3D world, objects have 3D shapes, and many applications need to have information about the length, width, and height of the object, as well as the deflection angle [23]. Camera-based 3D object detection methods mainly rely on extracting depth and spatial information from 2D images to achieve 3D detection, and estimating the 3D bounding box only from the image data input provided by the camera faces a great challenge since recovering 3D information from 2D input data is an unsettled problem. In the last two years, with the emergence of BEV perception, 3D object detection based on purely visual BEV schemes has attracted much attention, among which, the pioneering work of depth-based estimation, LSS [24], extracts the features of the surround-view camera image and "lifts" each image feature into a view cone. Then, all the view cones are "flattened" into a rasterized BEV grid to obtain the BEV feature map, and finally, the BEV feature map is processed using the task header to output the perceptual results. Inspired by LSS, subsequent works such as BEVDet were created [15]. Following the LSS paradigm, we propose a multi-angle camera 3D detection network architecture for use in BEV, which enhances the quality of depth estimation to enhance the performance of multi-view 3D object detection. The BEVDepth [25] approach, while maintaining the LSS structure, focuses on obtaining more accurate and optimized depth estimations and proposes a method to efficiently compute the voxel pooling process by introducing a multi-frame fusion technique. Utilizing this top-down BEV perspective can improve the performance of camera-based 3D detection algorithms by a large margin. However, camera-only-based methods usually need to rely on monocular depth estimation or stereo vision techniques due to insufficient depth information, but the depth estimation of these techniques is usually not accurate enough, which can lead to misdetection or omission, thus affecting the detection accuracy.

2.2. Three-Dimensional Object Detection Based on LiDAR

Current point cloud-based 3D object detection methods can be divided into two categories: voxel-based and pillar-based. Among the voxel-based methods, VoxelNet [23] is a 3D convolutional neural network-based algorithm that divides irregular point clouds into voxels and applies 3D convolutional aggregation of local voxel features, which are processed by a Region Proposal Network (RPN) to generate the object region of interest and predict the 3D bounding boxes. However, in the VoxelNet model, a large number of highly computational 3D convolutions are required, which makes real-time applications challenging. The subsequently proposed SECOND [13] network introduces 3D sparse convolution to accelerate and enhance the VoxelNet model in real time. In contrast to voxel-based methods, pillar-based approaches focus on reducing inference time and enhancing real-time performance. For example, the PointPillars [12] network utilizes pooling operations to transform point cloud features into pseudo-images in a bird's-eye view. It achieves end-to-end learning using only 2D convolutional layers, which significantly enhances the real-time performance of 3D object detection. Shi et al. from the Harbin Institute of Technology proposed the PillarNet [26] model, which introduces a 2D sparse convolution with a ResNet18 structure into the backbone of the BEV feature extraction module, in a way that improves real-time performance while being able to achieve similar accuracy to voxel-based networks [27].

2.3. Three-Dimensional Object Detection Based on Multi-Sensor Fusion

Due to certain limitations of individual sensors in tasks such as object detection and identification, the fusion of multiple sensors to maximize the benefits of each sensor is attracting more and more attention. Currently, there are numerous approaches based on camera and LiDAR fusion, such as MVAF-Net [28], PointFusion [29], RoarNet [30], DeepFusion [17], etc. MVAF-Net improves the performance of multi-view object detection in multi-view scenarios by introducing a view attention mechanism and feature fusion to process the input images from multiple viewpoints. PointFusion [29] processes point cloud data through multi-sensor fusion and feature fusion, thus enhancing the performance of 3D object detection and semantic segmentation tasks while effectively leveraging the information provided by various sensors. The core idea of RoarNet [30] is to first extract rough 3D object regions from LiDAR point clouds, and then use features from camera images to further refine these regions for object detection. Deepfusion [17] enhances the relationship between the image and LiDAR features during fusion by using InverseAug inverse geometric correlation enhancement as well as LearnableAlign fusion that utilizes cross-attention to capture correlations dynamically in order to obtain an effective alignment between multi-modal features. Despite the increasing maturity of these methods, there will still be the problem of object occlusion in complex scenes, whereas the application of BEV in object detection has a great advantage in that it provides a uniform and complete representation of the global scene, where the size and orientation of the objects can be directly represented. The steps for camera and LiDAR fusion in the BEV perspective include first extracting features from camera and LiDAR inputs and efficiently converting them into BEV features using view transformation. The transformed BEV features are then passed through a fully convolutional BEV encoder to obtain fused BEV features. Finally, these features are decoded to perform various sensing tasks. However, these fusion methods are highly dependent on multi-sensor data alignment, and inaccurate data alignment and synchronization can lead to a mismatch of multi-modal features in BEV representation, ultimately reducing the robustness and accuracy of object detection.

To address the above issues, the Att-BEVFusion model proposed in this paper predicts the depth distribution for each pixel in the camera view. Each feature pixel is then projected into multiple discrete points along the camera rays, with the associated features rescaled based on their respective depth probabilities. This process generates a feature point cloud, effectively spreading the features along the Z-axis. In the meantime, LIDAR point cloud voxelization is employed to convert the point cloud features into a uniform grid, and the associated convolution operation is performed to obtain the LIDAR BEV feature representation. In addition, we constructed a channel attention mechanism for fusing camera and LiDAR BEV features, along with a self-attention mechanism to enhance feature interaction. This approach strengthens the information exchange between the camera and LiDAR features as a way to improve detection accuracy.

3. General Structure

The overall structure of our proposed object detection for camera and LIDAR fusion in a BEV perspective is shown in Figure 2. The Att-BEVFusion algorithm is divided into four parts, which include (1) extracting the features from the camera inputs and transforming them into BEV features; (2) extracting the features from the LIDAR and transforming them into BEV features; (3) performing the features of the two under BEV fusion; and (4) inputting the object detection head to obtain the object detection result. To project both LiDAR and camera data into the BEV, the model performs transformations that account for the unique characteristics of each sensor type. Specifically, firstly, a 2D feature extractor is applied to extract features from the camera data, after which the extracted 2D features are transformed into 3D. These 3D features are then compressed to obtain the camera's BEV features. The processing of laser point cloud data is as follows: the 3D laser point cloud is first voxelized and then compressed to obtain the laser point cloud BEV features, and the transformed camera and LiDAR features are then fed into the channel attention mechanism, where the attention module dynamically adjusts the weights of the channel features that make a significant contribution to the task, amplifying the impact of these features so that they are more likely to be attended to and learned during subsequent processing. For the deep fusion features, we adopt the BEV self-attention mechanism to further enhance the interaction between different features, thus improving the expressive capability of the network when dealing with multi-scale information. We will test the effect of the attention module in later ablation experiments to assess its contribution to overall accuracy by comparing the performance of the Att-BEVFusion model with and without the attention mechanism.

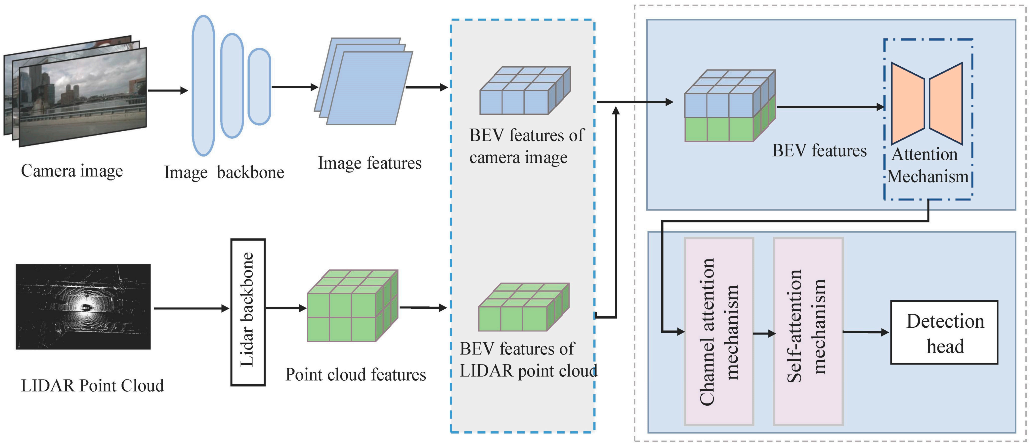


Figure 2. Overall structure diagram of Att-BEVFusion.

Our approach mainly consists of the following: 1. transformation of camera data to BEV space features; 2. transformation of LiDAR data to BEV space features; 3. feature fusion of the two in BEV space; 4. constructing the introduction of a channel attention mechanism and a self-attention mechanism; and 5. inputting the detector head.

3.1. Image Feature Extraction and Construction of BEV Features

The Att-BEVFusion fusion algorithm uses Resnet101 [31] as the backbone network to obtain rich semantic information and simultaneously introduces the feature pyramid network (FPN) [32], which is capable of extracting features from different scales using the pyramid structure and fusing them into a multi-scale feature representation that is suitable for detecting objects of different sizes. As shown in Figure 3, the input data is downsampled through three FPN blocks after passing through ResNet101 + FPN, producing feature maps $f_{1/8}$, $f_{1/16}$, and $f_{1/32}$. These feature maps are then upsampled using upsampling methods and 3 × 3 convolutional layers to unify their sizes to the same size as the 1/16 downsampled feature. This approach integrates multi-scale image features while preserving fine-grained information. Following this, average pooling downsampling and fully connected layers are applied, and the output feature map is obtained through a softmax function.

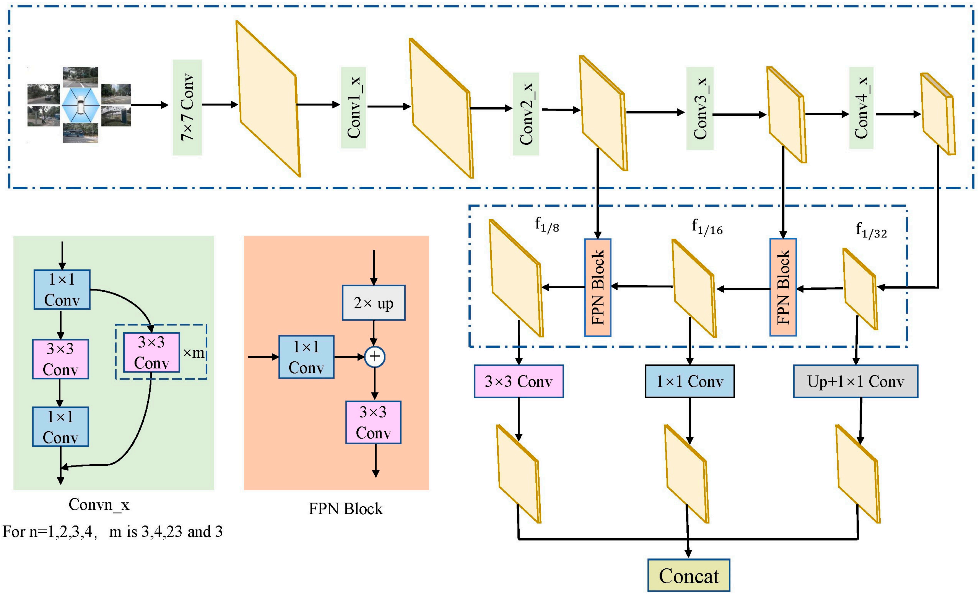


Figure 3. Extraction of image features.

The Att-BEVFusion fusion algorithm in this paper uses an implicitly supervised approach for camera BEV feature construction, which predicts the depth distribution for each pixel and projects the enriched image features to appropriate depth intervals in the 3D space, thus converting the image data into BEV features. Specifically, feature extraction is first performed on the input image data to obtain a high-dimensional feature map. The Lift–Splat–Shoot (LSS) method is applied to predict the depth distribution for each pixel in the camera view. Each feature pixel is then dispersed into multiple discrete points along the camera rays, with the corresponding features rescaled based on their depth probabilities to form a feature point cloud. Finally, the *Z*-axis is compressed in 3D space to generate the camera BEV features.

3.2. Transformation of LIDAR Features to BEV Features

The original LiDAR point cloud data contain rich depth information; however, due to the large volume of data, direct processing can impose a significant computational burden. To alleviate this burden, appropriate preprocessing of the point cloud data is necessary. When converting point cloud data to BEV features, it is common to compress the data along the *Z*-axis, which reduces the dimensionality and improves the efficiency of subsequent processing. The voxelization method for LiDAR point clouds provides a simplified and efficient representation of the data, accelerating the processing and feature extraction from the point clouds. In this paper, the Att-BEVFusion approach leverages the PointPillars [14] voxelization method to convert LiDAR point clouds into BEV features. As shown in Figure 4, specifically, it first divides the point cloud data into voxel pillars along the X and Y axes, resulting in P non-empty grids, each containing N point cloud data points. Features are extracted from each point cloud data point, and ultimately, all the point cloud features are aggregated into the voxel grid V_L. A Multi-Layer Perceptron (MLP) is then used to increase the dimensionality, producing $V_L^{'}$. Following this, a Softmax max-pooling operation is applied to reduce the dimensionality of the point cloud samples, resulting in a feature map with dimensions (C, P). The original coordinates are indexed based on the centers of the point cloud pillars, yielding a pseudo-image feature representation $V_L^{''}$ in the form of (C, H, W). Finally, a 2D backbone network is employed to elevate the dimensionality of the BEV features, obtaining high-dimensional BEV features from the LiDAR point clouds.

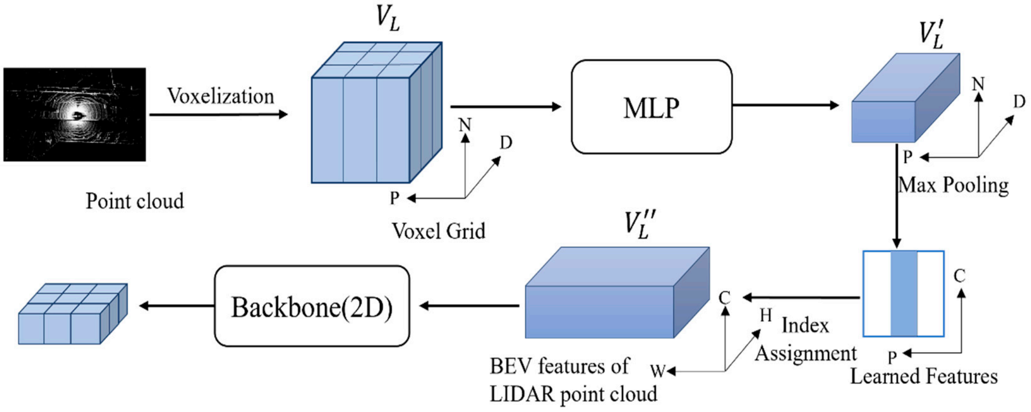


Figure 4. Transformation of LIDAR point cloud data to BEV features.

3.3. BEV Feature Fusion and Object Detection

3.3.1. Channel Attention Mechanism for Camera and LiDAR BEV Feature Fusion (CAM)

Figure 5 below shows the overall framework of the channel attention mechanism we introduced. The channel attention mechanism incorporates two main components: compression and excitation. The compression process includes two fully connected layers, followed by a ReLU activation function and a Softmax activation function. The features are first dimensionalized and then scaled, with the final weight vectors generated by a sigmoid function to ensure that they sum to one. First, the transformed image BEV features and LiDAR point cloud BEV features undergo global average pooling to downscale the feature values of each channel into a global vector. The attention layer will be weighted according to feature importance, prioritizing spatial accuracy from LiDAR data and semantic information from the camera to better adapt to the scene complexity and detection environment. Specifically, spatial feature compression is performed on the input feature map of dimension H × W × C, reducing it to a 1 × 1 × C feature map through global average pooling. Next, a 1 × 1 × C feature map with channel attention is learned via a fully connected layer. Finally, this feature map is multiplied channel by channel with the original input feature map (H × W × C) using the attention weights, producing the final output feature map with channel attention.

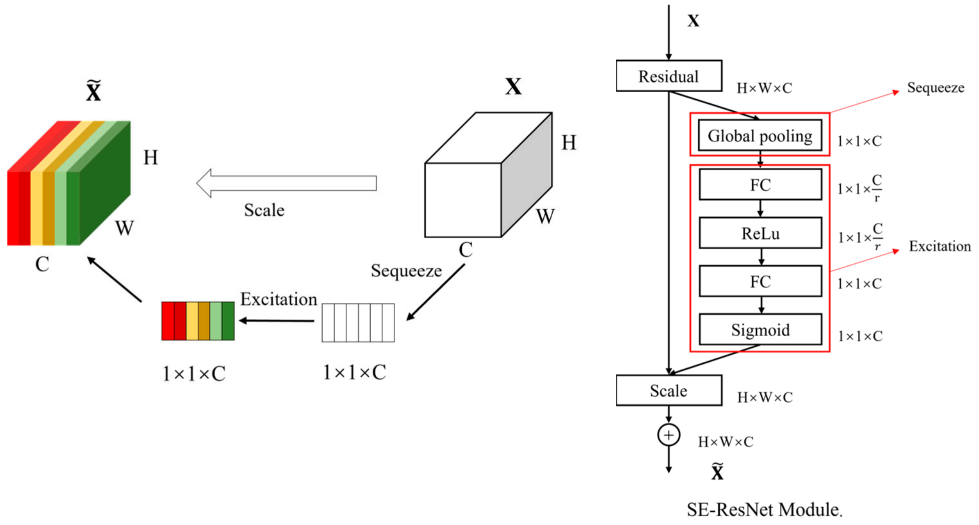


Figure 5. Structure of the channel attention mechanism (where r is the ratio of compression).

3.3.2. Self-Attention Mechanism of Feature Fusion (SAM)

Since the SE module [29] mainly focuses on inter-channel relationships, the fused feature maps ignore global information. To address the above problem, we introduce the BEV self-attention mechanism to globally operate on the features. This mechanism helps the fused features to infer their contextual positions in the overall BEV layout, thus aggregating information related to the shape of the object.

Figure 6 illustrates the overall structure of the BEV self-attention mechanism we constructed. Firstly, the BEV features are transformed into three components—query, key, and value—using linear transformations. The transposed key and query are then used for similarity calculations to obtain attention weights, which are subsequently normalized using the Softmax operation. Finally, the normalized weights are weighted and summed with the corresponding values to obtain the final self-attention feature map. The mathematical formulation of the self-attention mechanism is as follows:

$$Attention(q,k,v) = softmax\left(\frac{qk^T}{\sqrt{d}}\right)v \tag{1}$$

3.3.3. Three-Dimensional Object Detection

The fused BEV features combine the advantages of LiDAR point cloud BEV features and image BEV features, possessing both the rich semantic content of images and the 3D shape and structure information of LiDAR point clouds. This fusion helps to compensate for the limitations associated with using a single sensor. In this paper, we choose the object detection head in the PointPallars [14] method and feed the fused BEV features into the SSD [33] (Single Shot MultiBox Detector), which is a single-stage object detection algorithm based on convolutional neural networks, and it can directly carry out simultaneous object classification and localization, thus speeding up detection, while using multiple loss functions to simultaneously optimize object location and classification predictions, enabling efficient training of the neural network.

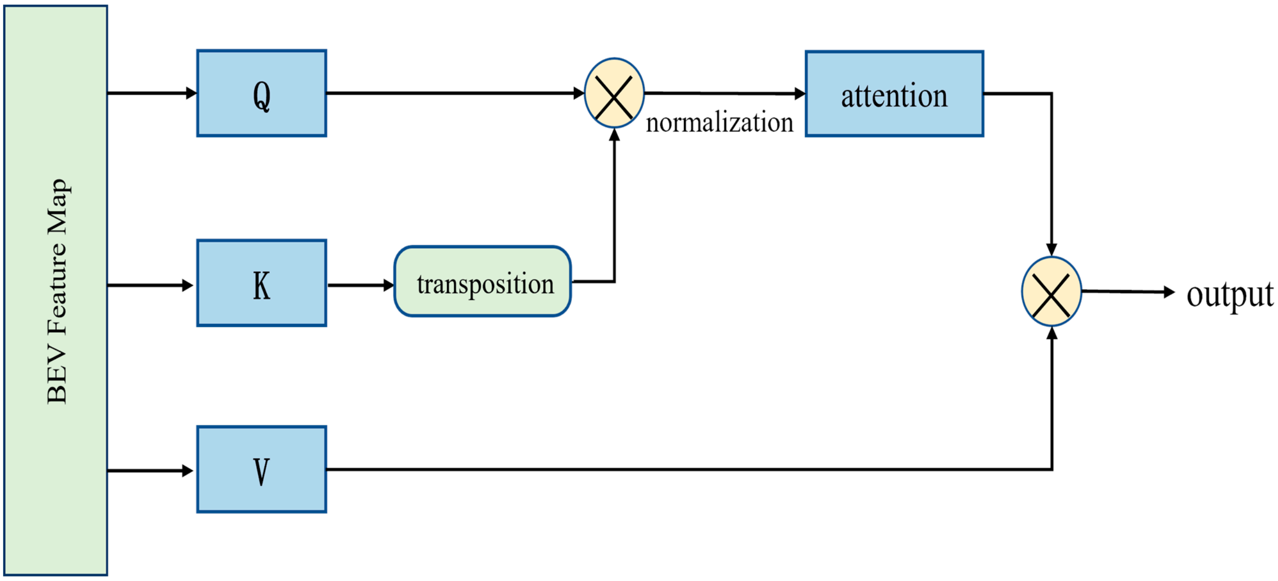


Figure 6. Structure of the self-attention mechanism.

3.4. Loss Function

The fusion algorithm in this paper is evaluated using three losses: object classification loss, 3D bounding box regression loss, and 3D bounding box orientation classification loss from the PointPillars [12] method.

First, for the object classification loss, we used the Focal Loss [31] loss function to achieve a balance of positive and negative samples and to determine the difficulty of classifying the samples:

$$L_{cls} = -\alpha_a (1 - p^a)^{\gamma} \ln(p^a) \tag{2}$$

where p^a represents the probability of the prediction box, α is 0.25, and γ is 2.0.

In the 3D bounding box regression loss task, the output bounding box is denoted as $(x, y, z, l, w, h, \theta)$, where x, y, z are the center coordinates of the bounding box, l, w, h are the dimensions of the bounding box, and θ is the angle of rotation of the bounding box. The regression residuals between the real bounding box and the predicted bounding box are defined as follows:

$$\Delta x = \frac{x^{gt} - x^a}{d^a}, \ \Delta y = \frac{y^{gt} - y^a}{d^a}, \ \Delta z = \frac{z^{gt} - z^a}{d^a} \tag{3}$$

$$\Delta w = ln\frac{w^{gt}}{w^a}, \ \Delta l = ln\frac{l^{gt}}{l^a}, \ \Delta h = ln\frac{h^{gt}}{h^a} \tag{4}$$

$$\Delta\theta = \theta^{gt} - \theta^a \tag{5}$$

Here, $*^{gt}$ denotes the true value, and $*^a$ denotes the predicted value. $d^a = \sqrt{(w^a)^2 + (l^a)^2}$. In this paper, the Smooth L1 loss function is used to calculate the geometric loss, which leads to the 3D bounding box regression loss L_{reg} of this paper:

$$L_{reg} = \sum\nolimits_{b \in (x,y,z,l,w,h,\theta)} SmoothL1(\Delta b) \tag{6}$$

Since the angular regression loss cannot distinguish the orientation, it will have some impact on the accuracy of the model. Therefore, we use the 3D bounding box orientation regression loss function L_{reg_θ} to solve this problem:

$$L_{reg_\theta} = SmoothL1\left(\sin\left(\theta^{gt} - \theta^{a}\right)\right) \tag{7}$$

where θ^{gt} denotes the true direction of the object, and θ^{a} denotes the predicted direction. When $\theta^{a} = \theta^{gt} \pm \pi$ time, the orientation regression loss tends to 0, which avoids the above situation and facilitates model training.

In order to solve the problem that the 3D object box orientation regression loss will treat the prediction boxes in opposite directions as the same, we use the cross-entropy loss function L_{dir} trained on the 3D bounding box orientation categories obtained from the 3D object prediction header to obtain more accurate orientation category prediction results:

$$L_{dir} = -\theta^{gt}_{dir} 1b\left(\theta^{a}_{dir}\right) - \left(1 - \theta^{gt}_{dir}\right) 1b\left(1 - \theta^{a}_{dir}\right) \tag{8}$$

where θ^{gt}_{dir} represents the orientation truth value, and θ^{a}_{dir} represents the predicted orientation category.

The final total loss function of the Att-BEVFusion algorithm consists of the above four loss functions:

$$L_{all} = \lambda_{cls} L_{cls} + \lambda_{reg}\left(L_{reg} + L_{reg_\theta}\right) + \lambda_{dir} L_{dir} \tag{9}$$

where λ_{cls}, λ_{reg}, and λ_{dir} are fixed loss weighting factors.

4. Experimentation

4.1. Datasets

We evaluated the proposed method on nuScenes [20], a shared large-scale dataset for automated driving. The dataset was collected from city streets in Singapore and Boston, USA, covering a variety of complex urban transportation scenarios, with 20 s long videos selected for each scenario, totaling about 15 h of driving data. The scenarios were selected with due consideration of diverse driving maneuvers and traffic situations and accidents, such as different locations, weather situations, vehicles traveling, driving rules, etc. Not only does it provide comprehensive annotations, but it also provides rich and diverse scenes and data for a variety of environment-aware tasks.

4.2. Evaluation Standards

We use the standard evaluation metrics of mAP (mean average precision) and detection score NDS (nuScenes detection score) for 3D object detection evaluation. mAP is a metric used to evaluate the overall performance, which calculates the performance evaluation of the 11 recall points and uses the mean average precision as the metric. NDS is a metric introduced in the NuScenes dataset to provide a more comprehensive evaluation of object detection models in autonomous driving. This score was designed to combine multiple aspects of performance, reflecting both the accuracy of the detected objects and the quality of their localization, which is particularly important in self-driving applications where precise positioning and classification of objects are crucial. NDS is calculated as follows. NDS is calculated as follows [20]:

$$NDS = \frac{1}{10}\left[5 \cdot mAP + 4 \cdot \left(1 - min\ (1,\ NDS_{L2})\right) + 1 \cdot \left(1 - min\ (1,\ NDS_{L1})\right)\right] \tag{10}$$

where mAP denotes mean accuracy, NDS_{L2} denotes the position error-based metric, and NDS_{L1} denotes attribute error-based metrics. Half of the NDS is based on mAP, while the other half evaluates the quality of the detections, taking into account metrics such as position, size, orientation, attributes, and velocity. In addition, we used the results of

10 detection categories for detailed comparison as a more comprehensive evaluation of 3D object detection results.

4.3. Experimental Details

We tested the network on MMDetection3D [34], a PyTorch [35]-based object detection library, which is among the most popular toolkits in the realm of object detection due to its highly encapsulated mechanism. For the image branch, we used ResNet101 as the image backbone; the ResNet101 architecture enables the extraction of more comprehensive and rich semantic features due to its depth and complexity. For the LiDAR branch, the raw point cloud can be processed using the Pointpillars [14] point cloud voxelization method. The Pointpillars method forms a pseudo-image by dividing the point cloud into vertical columnar regions of fixed size, mapping the features extracted in the region to a 2D space, and finally extracting high-level features on the pseudo-image, a step similar to that used in traditional methods. This step is similar to object detection in traditional image processing and effectively reduces computational complexity.

For our experiments, we applied FPN to fuse multi-scale camera features to generate feature maps of 1/16 input size, with voxel sizes of LiDAR point clouds set to 0.075 m, 0.075 m, and 0.2 m according to the experimental setup. Our training and inference were performed on an Ubuntu 18.04 server with an I7-10700 CPU and GeForce RTX 3060 GPU. The development language used for the experiments was Python 3.7, based on the Pytorch deep learning structure to write the model code. The model code was written using the AdamW [36] optimizer to optimize the parameters of the network with a learning rate of 2×10^{-4} and a weight decay of 1×10^{-2}.

4.4. Test Results and Comparison

To thoroughly evaluate the performance of the Att-BEVFusion approach on the nuScenes dataset, we compared the detection results of Att-BEVFusion with other advanced methods. As indicated in Table 1, these methods are categorized into camera-based, LiDAR-based, and LiDAR and camera fusion-based. BEV camera-based methods only, such as BEVDet [15], BEVFormer [37], and BEVHeight [38], LIDAR-based methods only, such as CenterPoint [39], Deeproute [40], and TransFusion-L [41], and camera and LIDAR fusion-based methods, such as FusionPainting [42], PointAugmenting [2], TransFusion [41], BEVFusion [43], and BEVFusion4D [44], are compared. Our proposed Att-BEVFusion algorithm achieves 72.0% mAP and 74.3% NDS. According to Table 1, our testing method surpasses previous state-of-the-art approaches in most testing categories, and for the more challenging pedestrian (Pedestrian) and bicycle (Bicycle) categories, respectively, Att-BEVFusion achieved competitive mAPs of 91.8% and 60.0%, which outperformed all single-sensor and multi-sensor fusion methods, a result that also demonstrates the positive impact of our proposed fusion method on object detection. There are also a few methods whose performance lags behind slightly, and this performance difference can be attributed to several factors including, but not limited to, variations in the experimental setup, training process, and dataset partitioning. And since our results were obtained under specific conditions that may be different from the studies given in BEVFusion4D [44] and BEVFusion [19], this may lead to differences in the results. It should be noted that the table also reports the runtime per frame, and compared to the camera-only and lidar-only approaches, our Att-BEVFusion requires additional runtime due to generating the camera features in BEV space, so our processing is a little bit slower, although the overall performance is still good. The overall performance improvement is attributed to the introduction of the channel attention mechanism and the BEV self-attention mechanism for fusing LiDAR and camera BEV features.

Table 1. This shows the evaluation results for the nuScenes test set. L denotes a LIDAR-based method. C denotes a camera-based method. L + C denotes a LIDAR–camera-based method. Abbreviations stand for construction vehicle (C.V.), motorcycle (Motor.), pedestrian (Ped.), and traffic cone (T.C.). Red and blue colors represent the optimal and sub-optimal results, respectively.

Methods	Modality	mAP	NDS	Latency (ms)	Car	Truck	C.V.	Bus	Trailer	Barrier	Motor.	Bicycle	Ped.	T.C.
BEVDet [15]	C	42.4	47.6	-	64.3	35.0	16.2	35.8	35.4	61.4	44.8	29.6	41.1	60.1
BEVFormer [37]	C	48.1	56.9	-	67.7	39.2	22.9	35.7	39.6	62.5	47.9	40.7	54.4	70.3
BEVHeight [38]	C	53.2	61.0	-	68.6	44.8	27.4	42.8	48.5	69.8	54.2	45.9	57.7	72.6
CenterPoint [39]	L	60.3	67.3	80.7	85.2	53.5	20.0	63.6	56.6	71.1	59.5	30.7	84.6	78.4
Deeproute [40]	L	60.6	68.1	-	82.9	51.5	25.1	59.5	47.6	65.2	68.6	44.3	84.4	76.4
TransFusion-L [41]	L	65.5	70.2	-	86.3	56.7	28.1	66.2	58.7	78.0	68.4	44.2	86.2	82.0
3D-CVF [45]	L + C	52.7	62.4	-	83.3	45.1	15.7	48.6	49.5	65.7	51.2	30.6	74.1	62.9
FusionPainting [42]	L + C	68.1	72.0	-	87.1	60.8	30.0	68.5	61.7	71.8	74.7	53.5	88.3	85.0
TransFusion [41]	L + C	68.9	71.5	156.6	87.5	59.9	33.0	68.1	60.9	78.1	73.5	52.9	88.4	86.7
BEVFusion [43]	L + C	70.2	72.3	-	88.6	60.1	**39.3**	69.8	63.8	80.0	74.1	51.0	89.2	86.5
DeepInteraction [46]	L + C	70.8	73.7	-	87.7	60.4	37.9	70.6	63.8	**80.4**	75.4	54.5	**91.7**	**87.2**
BEVFusion [19]	L + C	71.3	73.4	119.2	88.3	**70.0**	34.3	69.1	62.1	78.5	72.1	52.0	89.2	86.7
BEVFusion4D [44]	L + C	**71.9**	**73.7**	-	**88.8**	64.0	**38.0**	**72.8**	**65.0**	79.8	**77.0**	**56.4**	90.4	87.1
Ours	L + C	**72.0**	**74.3**	141.3	**88.9**	**64.8**	30.2	**73.5**	**64.2**	**80.0**	**78.9**	**60.0**	**91.8**	**87.7**

4.5. Ablation Experiments

To validate the efficiency and reasonableness of the designed approach module, ablation experiments were conducted for camera view to BEV transformation (CBT), channel attention mechanism (CAM) for camera and lidar BEV feature fusion, and self-attention mechanism (SAM) for feature fusion. To shorten the time of the experiment and increase the efficiency of 3D object detection, we utilized 1/4th of the training data of the nuScenes dataset for the training and testing of the entire ablation experiments. Here, the average accuracy mAP and the detection score NDS of 3D object detection are used as metrics of approach performance, and the approach is evaluated with the baseline network. The algorithm is assessed in comparison with the baseline network. As shown in Table 2, which demonstrates the change in the performance of the network after adding different components, the optimal values in each experiment are shown in boldface, where the baseline indicates a voxelized detection structure based on LiDAR only, without point cloud and image fusion.

Table 2. Contribution of each module to the network.

	Baseline	CBT	CAM	SAM	mAP	NDS	Car	Bicycle	Ped.
a	√	--	--	--	58.4	66.5	84.0	54.3	83.1
b	√	√	--	--	59.2	68.2	85.2	55.6	83.6
c	√	√	√	--	61.3	69.6	86.6	56.2	84.1
d	√	√	√	√	**62.3**	**70.1**	**87.2**	**58.6**	**85.5**

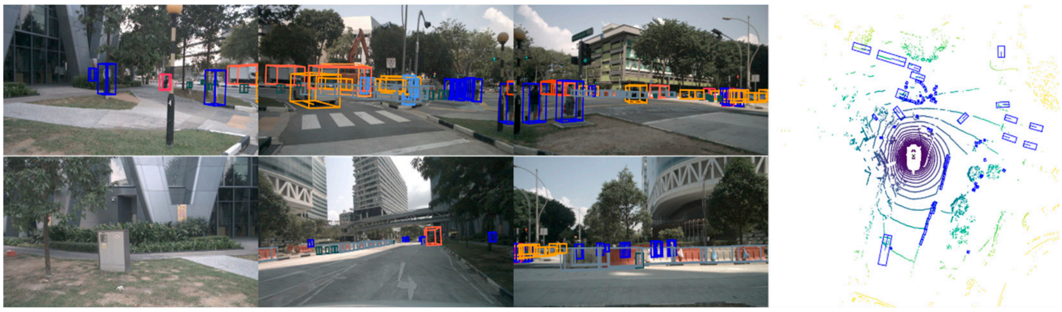
4.5.1. Quantitative Analyses

As can be seen in Table 2, when the CBT module is introduced into the LiDAR-only-based detection structure, all the mAP and NDS values are increased, which demonstrates the effectiveness of our proposed fusion approach. Particularly in terms of the accuracy of 3D object detection, some improvement is achieved: for specific categories, the car, bicycle, and pedestrian categories are improved by 1.2%, 1.3%, and 0.5%, respectively, which is due to the fact that LiDAR point cloud data is sparse and low resolution, containing less valid information, which leads to the pure point cloud detection framework is difficult to have a better performance for the detection effect of 3D objects, the more such targets need to be supplemented by image information. After adding the CAM module, the network can select and weight the important features in each channel more effectively, this enhances the effectiveness of feature fusion and the quality of image BEV features, resulting in a significant improvement in the detection accuracy presented in this paper. The car class, bicycle class, and pedestrian class are improved by 1.4%, 0.6%, and 0.5%, respectively, in which the improvement is larger for the car class, but the improvement is not obvious in the detection of small object objects such as bicycle class and pedestrian class, which may be due to the lack of global inference in the BEV fusion feature context generated by the introduction of the CAM, and the features distributed in different locations cannot fully interact with each other. The BEV fusion features can only provide local information but cannot provide global integrated reasoning, resulting in an insignificant improvement of small object detection accuracy. After adding the SAM module, it is able to better grasp the global information and generate feature maps that are more in line with the object location distribution of the real scene, with 0.6%, 2.4%, and 1.4% enhancement for the car class, bicycle class, and pedestrian class, respectively; compared to the baseline, our method improves the mAP and NDS detection scores by 3.9% and 3.6%, respectively. This part of the ablation experiments demonstrates the effectiveness of the modules of the network architecture in this paper.

4.5.2. Qualitative Analysis

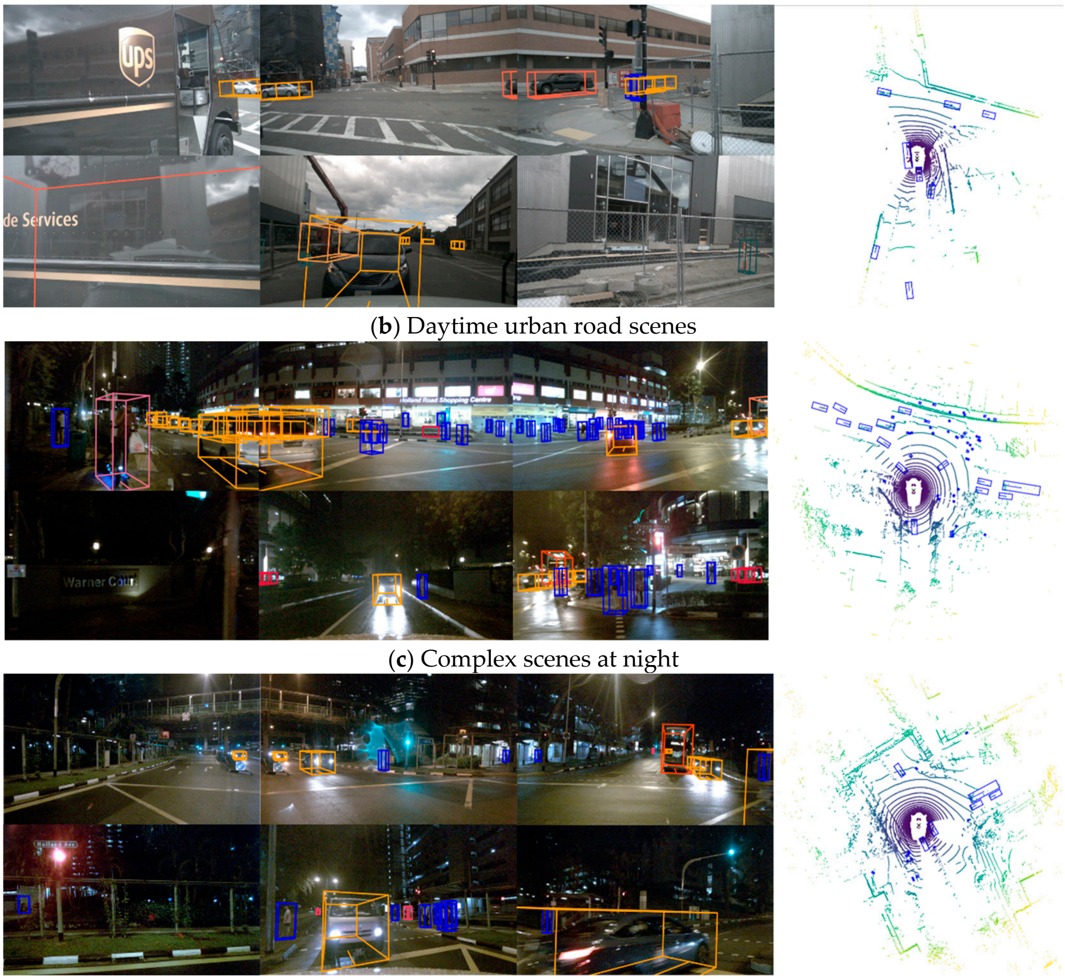
To demonstrate the effectiveness of the proposed method, we selected object detection results from six images in the nuScenes test set for visualization including daytime and nighttime urban roads, as well as complex conditions such as intersections, to highlight the superiority of our method. As shown in Figure 7. From Figure 7a,b, it can be seen that our method performs well even in high-traffic and densely populated areas, such as city streets and intersections during the daytime. It also effectively detects pedestrians and smaller objects, such as cyclists and scooter riders, and in Figure 7c,d, it can be seen that, for vehicles in the nighttime traffic environment with complex traffic environments and congested vehicles and pedestrians, the Att-BEVFusion algorithm is still able to carry out accurate recognition and differentiation, accurately identifying multiple objects in front of it, with good adaptive ability and anti-interference ability. This suggests that the organic combination of the channel attention mechanism and the BEV self-attention mechanism enables our method to make full use of the depth-interacting BEV fusion features to effectively detect occluded objects.

These results highlight that our approach not only achieves great improvement in detection performance but also still demonstrates convincing reliability when dealing with challenges in complex scenes.



(**a**) Daytime junction scene

Figure 7. *Cont.*



(**b**) Daytime urban road scenes

(**c**) Complex scenes at night

(**d**) Nighttime junction scene

Figure 7. Att-BEVFusion qualitative detection results.

5. Conclusions

In this paper, we propose a 3D object detection method (Att-BEVFusion) based on BEV fusion of the camera and LiDAR. By effectively fusing the camera view features and LiDAR point cloud features in BEV space, and combining this with the channel attention mechanism and the self-attention mechanism, it significantly improves the accuracy and robustness of object detection. The experimental results show that Att-BEVFusion achieves 72.0% mAP and 74.3% NDS on the nuScenes dataset, and 88.9% and 91.8% accuracy in car and pedestrian detection tasks, respectively, which fully proves the superiority of the method in multi-sensor fusion.

Meanwhile, the Att-BEVFusion method demonstrates significant advantages in long-range object and small object detection, especially in complex scenarios where it still maintains high accuracy and robustness. The method in this paper provides a reliable multi-sensor fusion solution for autonomous driving systems and provides new research directions for 3D object detection tasks.

In our study, the phenomenon of overlapping targets has an impact on the accuracy of the data. However, due to time and resource constraints, we did not deal with the problem in depth in our current work. Therefore, future research could further explore effective methods to cope with target overlapping. Secondly, we will consider researching more complex traffic scenes and higher-density traffic targets to improve the generalization ability of the algorithm; in addition, we will explore more multi-modal data fusion strategies, such as incorporating radar or ultrasonic sensors, to further improve the detection accuracy and robustness in complex environments.

In conclusion, Att-BEVFusion provides effective support for the development of autonomous driving perception systems, and its robustness and high-precision detection capability lay a solid foundation for the security and stability of autonomous driving. Future research can be optimized based on this method to promote the further application and development of autonomous driving technology.

Author Contributions: Conceptualization, M.Z. and P.S.; data curation, M.Z. and A.Y.; formal analysis, M.Z. and X.D.; investigation, P.S.; methodology, M.Z. and P.S.; project administration, A.Y.; software, X.D.; visualization, M.Z.; writing—original draft, M.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Central Guided Local Science and Technology Special Project–Yangtze River Delta Science and Technology Innovation Community Joint Tackling Program Project (2023CSJGG1600); the Natural Science Foundation of Anhui Province (2208085MF173); and the Wuhu City "Red Casting Light" Major Science and Technology Project (2023zd01, 2023zd03).

Data Availability Statement: The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Chen, W.; Zhou, C.; Shang, G.; Wang, X.; Li, Z.; Xu, C.; Hu, K. SLAM overview: From single sensor to heterogeneous fusion. *Remote Sens.* **2022**, *14*, 6033. [CrossRef]
2. Li, H.; Sima, C.; Dai, J.; Wang, W.; Lu, L.; Wang, H.; Zeng, J.; Li, Z.; Yang, J.; Deng, H. Delving into the devils of bird's-eye-view perception: A review, evaluation and recipe. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *46*, 2151–2170. [CrossRef] [PubMed]
3. Vora, S.; Lang, A.H.; Helou, B.; Beijbom, O. Pointpainting: Sequential fusion for 3d object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 4604–4612.
4. Wang, C.; Ma, C.; Zhu, M.; Yang, X. Pointaugmenting: Cross-modal augmentation for 3d object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 11794–11803.
5. Chen, X.; Ma, H.; Wan, J.; Li, B.; Xia, T. Multi-view 3d object detection network for autonomous driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1907–1915.
6. Ku, J.; Mozifian, M.; Lee, J.; Harakeh, A.; Waslander, S.L. Joint 3d proposal generation and object detection from view aggregation. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1–8.
7. Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; Zhang, L. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Trans. Image Process.* **2017**, *26*, 3142–3155. [CrossRef] [PubMed]
8. Pang, S.; Morris, D.; Radha, H. CLOCs: Camera-LiDAR object candidates fusion for 3D object detection. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October 2020–24 January 2021; pp. 10386–10393.
9. Pang, S.; Morris, D.; Radha, H. Fast-CLOCs: Fast camera-LiDAR object candidates fusion for 3D object detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2022; pp. 187–196.
10. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
11. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5105–5114.
12. Zhou, Q.; Yu, C. Point rcnn: An angle-free framework for rotated object detection. *Remote Sens.* **2022**, *14*, 2605. [CrossRef]
13. Yan, Y.; Mao, Y.; Li, B. Second: Sparsely embedded convolutional detection. *Sensors* **2018**, *18*, 3337. [CrossRef] [PubMed]
14. Lang, A.H.; Vora, S.; Caesar, H.; Zhou, L.; Yang, J.; Beijbom, O. Pointpillars: Fast encoders for object detection from point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 12697–12705.
15. Huang, J.; Huang, G.; Zhu, Z.; Ye, Y.; Du, D. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv* **2021**, arXiv:2112.11790.
16. Huang, J.; Huang, G. Bevpoolv2: A cutting-edge implementation of bevdet toward deployment. *arXiv* **2022**, arXiv:2211.17111.
17. Li, Y.; Yu, A.W.; Meng, T.; Caine, B.; Ngiam, J.; Peng, D.; Shen, J.; Lu, Y.; Zhou, D.; Le, Q.V. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 17182–17191.

18. Li, M.; Zhang, Y.; Ma, X.; Qu, Y.; Fu, Y. BEV-DG: Cross-Modal Learning under Bird's-Eye View for Domain Generalization of 3D Semantic Segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 11632–11642.
19. Liu, Z.; Tang, H.; Amini, A.; Yang, X.; Mao, H.; Rus, D.L.; Han, S. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA), London, UK, 29 May–2 June 2023; pp. 2774–2781.
20. Caesar, H.; Bankiti, V.; Lang, A.H.; Vora, S.; Liong, V.E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; Beijbom, O. nuscenes: A multimodal dataset for autonomous driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11621–11631.
21. Shi, P.; Liu, Z.; Dong, X.; Yang, A. CL-fusionBEV: 3D object detection method with camera-LiDAR fusion in Bird's Eye View. *Complex Intell. Syst.* **2024**, *10*, 7681–7696. [CrossRef]
22. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
23. Ma, Y.; Wang, T.; Bai, X.; Yang, H.; Hou, Y.; Wang, Y.; Qiao, Y.; Yang, R.; Manocha, D.; Zhu, X. Vision-centric bev perception: A survey. *arXiv* **2022**, arXiv:2208.02797. [CrossRef] [PubMed]
24. Philion, J.; Fidler, S. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV*; Springer: Cham, Switzerland, 2020; pp. 194–210.
25. Li, Y.; Ge, Z.; Yu, G.; Yang, J.; Wang, Z.; Shi, Y.; Sun, J.; Li, Z. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington DC, USA, 7–14 February 2023; pp. 1477–1485.
26. Shi, G.; Li, R.; Ma, C. Pillarnet: Real-time and high-performance pillar-based 3d object detection. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2022; pp. 35–52.
27. Shi, P.; Dong, X.; Yang, A. Research Progress on Bev Perception Algorithms for Autonomous Driving: A Review. Available online: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4790559 (accessed on 21 October 2024).
28. Wang, G.; Tian, B.; Zhang, Y.; Chen, L.; Cao, D.; Wu, J. Multi-view adaptive fusion network for 3D object detection. *arXiv* **2020**, arXiv:2011.00652.
29. Xu, D.; Anguelov, D.; Jain, A. Pointfusion: Deep sensor fusion for 3d bounding box estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 244–253.
30. Shin, K.; Kwon, Y.P.; Tomizuka, M. Roarnet: A robust 3d object detection based on region approximation refinement. In Proceedings of the 2019 IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019; pp. 2510–2515.
31. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
32. Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In *Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; Honolulu, HI, USA, 21–26 July 2017, pp. 2117–2125.
33. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I*; Springer: Cham, Switzerland, 2016; pp. 21–37.
34. MMDetection3D Contributors. OpenMMLab's Next-Generation Platform for General 3D Object Detection. 2020. Available online: https://openmmlab.medium.com/mmdetection3d-the-next-generation-3d-object-detection-platform-8a17d9292d3c (accessed on 21 October 2024).
35. Imambi, S.; Prakash, K.B.; Kanagachidambaresan, G. PyTorch. In *Programming with TensorFlow: Solution for Edge Computing Applications*; Springer: Cham, Switzerland, 2021; pp. 87–104.
36. Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. *arXiv* **2017**, arXiv:1711.05101.
37. Li, Z.; Wang, W.; Li, H.; Xie, E.; Sima, C.; Lu, T.; Qiao, Y.; Dai, J. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2022; pp. 1–18.
38. Yang, L.; Yu, K.; Tang, T.; Li, J.; Yuan, K.; Wang, L.; Zhang, X.; Chen, P. Bevheight: A robust framework for vision-based roadside 3d object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 21611–21620.
39. Yin, T.; Zhou, X.; Krahenbuhl, P. Center-based 3d object detection and tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 11784–11793.
40. Han, J.; Sun, A. DeepRouting: A deep neural network approach for ticket routing in expert network. In Proceedings of the 2020 IEEE International Conference on Services Computing (SCC), Beijing, China, 7–11 November 2020; pp. 386–393.
41. Bai, X.; Hu, Z.; Zhu, X.; Huang, Q.; Chen, Y.; Fu, H.; Tai, C.-L. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 1090–1099.

42. Xu, S.; Zhou, D.; Fang, J.; Yin, J.; Bin, Z.; Zhang, L. Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection. In Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 19–22 September 2021; pp. 3047–3054.
43. Liang, T.; Xie, H.; Yu, K.; Xia, Z.; Lin, Z.; Wang, Y.; Tang, T.; Wang, B.; Tang, Z. Bevfusion: A simple and robust lidar-camera fusion framework. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 10421–10434.
44. Cai, H.; Zhang, Z.; Zhou, Z.; Li, Z.; Ding, W.; Zhao, J. BEVFusion4D: Learning LiDAR-Camera Fusion Under Bird's-Eye-View via Cross-Modality Guidance and Temporal Aggregation. *arXiv* **2023**, arXiv:2303.17099.
45. Yoo, J.H.; Kim, Y.; Kim, J.; Choi, J.W. 3d-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII*; Springer: Cham, Switzerland, 2020; pp. 720–736.
46. Yang, Z.; Chen, J.; Miao, Z.; Li, W.; Zhu, X.; Zhang, L. Deepinteraction: 3d object detection via modality interaction. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 1992–2005.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.