2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

LSSAttn: Towards Dense and Accurate View Transformation for Multi-modal 3D Object Detection

Qi Jiang[1] and Hao Sun[2]

***Abstract*— Fusing the camera and LiDAR information in the unified BEV representation serves as the elegant paradigm for the 3D detection tasks. Current multi-modal fusion methods in BEV can be categorized into LSS-based and Transformer-based in terms of their view transformation. The former leverages inaccurate depth prediction and massive pseudo points for perspective-to-BEV transformation while the latter only fetches sparse image features to the BEV representation. To overcome their shortcomings, an optimized view transformation is proposed, which can be easily modulated into the LSS-based methods. The proposed module capitalizes on the LSS mechanism to establish dense associations between perspective pixels and BEV grids. It utilizes the attention mechanism to compute similarity scores for each associated pair during feature aggregation. Starting from the BEVFusion baseline, we further introduce (1) cross-attention within the associated subsets to transfer image features into the BEV, and (2) a multi-scale feature fusion mechanism for LSS-based view transformation. Extensive experiments on nuScenes validate the effectiveness and efficiency of our proposed module, which achieves an increase of 1.3% in mAP compared to the baseline model.**

I. INTRODUCTION

Autonomous vehicles are usually equipped with various sensors to perceive the surrounding environments[1]. LiDAR and cameras are the most common sensors in autonomous driving due to their complementary characteristics, resulting in significant performance improvements in fusion methods. The substantial disparities in data formats from different sensors necessitate careful cross-modal fusion strategies. Current state-of-the-art (SOTA) methods[2], [3], [4] follow the trend to transform sensor information from distinct modalities into a unified bird's eye view (BEV) representation, where the fused feature map is employed for various downstream 3D tasks. LiDAR-derived point clouds in 3D space readily facilitate the acquisition of BEV feature maps. Conversely, view transformation (VT) from the image plane to the BEV is necessary for image information.

Inspired by image-based BEV perception[5], [6], multi-modality BEV perception mainly varies in the view transformation for image feature, which can be classified into two main streams: LSS-based (Lift-Splat-Shoot)[7] and Transformer-based[6]. Previous works on BEV perception[8], [6], [2], [3] neglect the deep investigation of the functional essence of the two mainstream view transformation modules as well as the fair comparisons

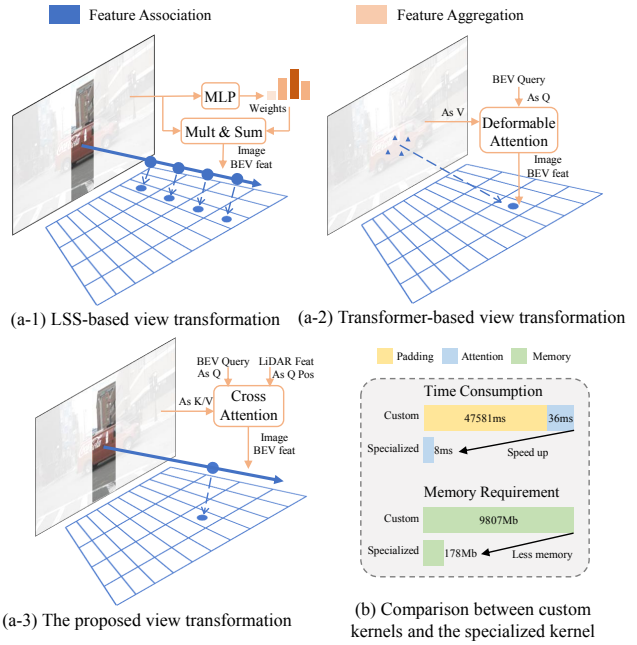


Fig. 1. **View transformation comparisons.** (a-1) LSS-based VT leverages 3D pseudo points for the dense feature association and utilizes the MLP for predicted weights to aggregate features. (a-2) The transformer-based VT uses predefined 3D reference points to fetch image features and capitalizes on the attention mechanisms to aggregate corresponding features. (a-3) The proposed VT is the combination of the LSS-based dense feature association and attention-based feature aggregation. (b) The optimized kernel enables the proposed VT with less time and memory.

of their efficiency and effectiveness. In BEV pooling, the content vectors within the same pillars are weighted with depth score and added, while the similarity is calculated to weigh and sum the value in cross-attention. Therefore, we believe that the current mainstream view transformations are essentially consistent and can be decomposed into two steps that contribute to the performance boost:

- **Feature association** to establish the associations between perspective pixels and BEV grids. LSS-based[7], [2], [3] methods leverage the calibration parameters to generate 3D pseudo points, allowing for the dense associations. Transformer-based methods[4] leverage reference points to establish associations between sparse perspective pixels and BEV grid.
- **Feature aggregation** by computing weights to aggregate the associated image features as BEV features. In LSS-based methods, the depth distribution predicted

[1]Equal contributions. Qi Jiang is with Shanghai Jiao Tong University. Work done during the internship in BOSCH Corporate Research. kiki_jiang@sjtu.edu.cn
[2]Equal contributions. Hao Sun is with BOSCH Corporate Research. sun_hao@u.nus.edu

from images serves as weights. In transformer-based methods, each layer of the Transformer block essentially employs attention mechanisms to calculate similarity scores, acting as weights.

These two view transformation modules emphasize distinct aspects, with unique strengths in feature association and feature aggregation. The feature association of LSS enables dense supervision signals backward to perspective features while the gradients are not accurately adjusted due to the coarse depth prediction. The cascaded Transformer facilitates proper feature aggregation while supervision signals on perspective features are too sparse[6]. To enhance the effectiveness and efficiency of the view transformation module, we propose LSSAttn, leveraging the advantages of both approaches. The proposed module capitalizes on the LSS mechanism to establish dense associations and the attention mechanism to compute similarity scores for each associated pair. However, the dense feature association presents a challenge for attention-based similarity in computation complexity and memory consumption when padding the repeated features into standard inputs for cross-attention as in Fig.I-(b). Here, we propose to perform cross-attention within the association subsets, which divide the repeated camera perspective features and LiDAR BEV features into subsets and access the subsets feature by index. We introduce an optimized GPU-accelerated operator to enable cross-attention within each subset.

With the combination of the dense associations from LSS[7] and the efficiency of attention mechanisms, LSSAttn optimally integrates image and LiDAR information within the BEV domain. Our contributions can be summarized as follows: (1) We revisit and delve into the view transformation modules in multi-modal BEV perception and provide a deep understanding of the mechanism of the view transformation. (2) We propose an optimized view transformation, namely LSSAttn to combine the strengths of previous VT modules and introduce multi-scale feature aggregation for LSS-based methods. (3) Experiments are conducted to confirm the effectiveness of the proposed module.

II. RELATED WORK

A. Camera-based 3D detection

Early works deriving from monocular 2D detection extend extra 3D attribute regression branches, including dimensions, observation angle, and center depth[12], [13]. To mitigate the gap between 3D detection and 2D feature maps, current trends pursue the unified BEV representation of the surround-view cameras. The view transformation transfers separate perspective-view features into the BEV, enabling the unified and suitable feature representation for 3D tasks. Following LSS[7], methods like BEVDet[8] and BEVDepth[5] leverage calibration to lift and splat dense image features into BEV. Methods like fast-BEV[14] put emphasis on the deployment-friendly design of the BEV model. SimpleBEV[15] conducts comprehensive experiments to elucidate the high-impact factors in the design and training of the BEV perception model and proposes the pull-style lifter for view transformation. Other methods define BEV queries and use fixed reference points and cascaded transformers to fetch the sparse image features[6], [16], [17].

B. LiDAR-based 3D detection

Methods like PointNet[18], PointFormer[19], and PointRCNN[20] directly take points as input to obtain points-level features. Due to the irregular and unordered characteristics of LiDAR points, other methods tend to firstly discrete the points into regular and ordered representations like pillars[21] and voxels[22], then apply standard 2D or 3D convolution backbone for feature extraction. SECOND[11] optimizes the backbone efficiency with sparse convolution. DSVT[23] leverages the high-performance transformer to dynamically encode sparse voxel features. Following 2D detection [24], [25], anchor-based 3D detection head is commonly used while CenterPoint [26] and TransFusion[27] adopt a center-based representation for anchor-free detection head. VoxelNeXt[28] introduces the fully sparse VoxelNet[22] with MLP or sparse 3D convolution as detection head.

C. Multi-modal 3D Detection

The complementary characteristics of the LiDAR and camera have made fusion a heated topic in the 3D detection community. Early approaches employed proposal-level fusion by either projecting 3D proposals to image plane[29] or lifting 2D proposals into the 3D world to form the region of interest[30], highly sensitive to the proposal quality. Point-level fusion decorates LiDAR points with the feature-level[31] or result-level[32], [33] camera information and modulates the augmented points into the off-the-shelf LiDAR-based detection models[21], [22], [26], [27]. It achieves a high-performance boost with minor modifications to the model architecture. However, the sparse association established with the projected LiDAR points leaves image features partially exploited. Following the trend in 2D detection, recent methods pursue feature-level fusion in the unified representation. Methods like BEVFusion[2], [3] follow BEVDet[8], which leverage LSS[7] to transform image features from a perspective view to BEV. SemanticBEV[34] introduces explicit semantic cues into the fused BEV features. EA-LSS[35] refines the LSS mechanism with an edge-aware depth map. Inspired by BEVFormer[6], FUTR3D[4] define the query in the unified BEV representation and use reference points to fetch features from different modalities, including LiDAR, images, and radars. BEVFusion4D[36] uses the LiDAR feature as the query to guide the view transformation. These two trends mainly differentiate from each other in the view transformation, which establishes associations between perspective pixels and BEV grids and aggregates corresponding image features.

III. METHODOLOGY

In this section, the detailed architecture of the LiDAR-camera fusion for 3D detection will be presented. As shown

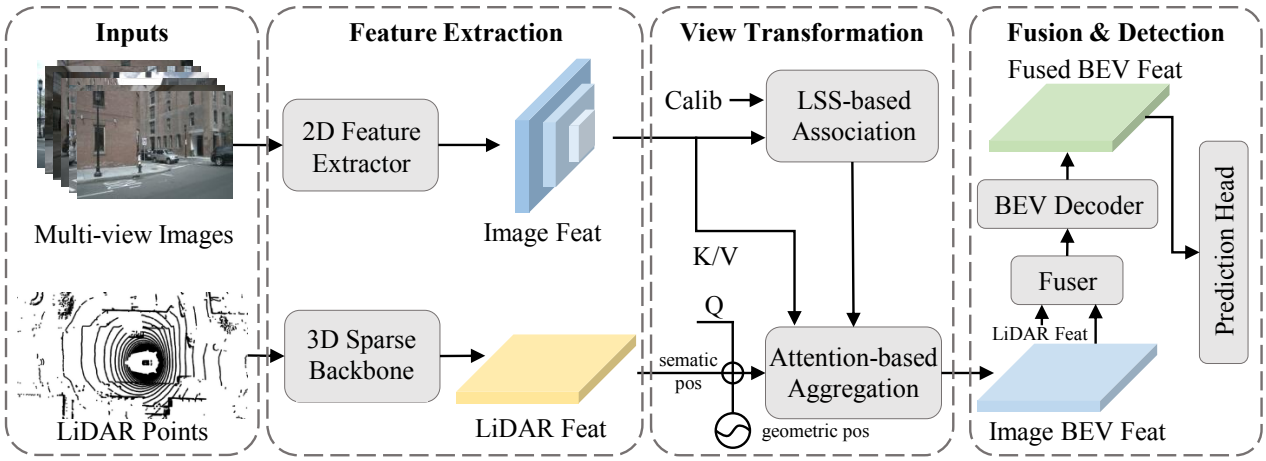


Fig. 2. **An overview of the LSSAttn pipeline.** The surround-view images and LiDAR points are separately input to the 2D feature extractor (e.g. Swin-tiny[9] and FPN[10]) and 3D sparse backbone (e.g. SparseEncoder[11]). The VT module first establishes dense associations between perspective pixels and BEV grids and then aggregates image features as BEV features with cross-attention, where image features serve as both key and value, while LiDAR BEV features act as semantic query positional encoding. The VT module leads to the creation of image BEV features that are aligned with their LiDAR counterparts. These unified multi-modal features are then seamlessly integrated through convolutional fusing and BEV decoding, resulting in robust fused BEV features where task-specific heads are further employed to accomplish 3D detection tasks.

in Fig.2, our model follows the customary design of previous multi-modal BEV perception models[2], [3] with two separate feature extraction branches, a view transformation module and a shared BEV fuser and detection head.

A. Feature Extraction

We follow the paradigm in previous work[2], [3] and adopt a dual-branch feature extraction pipeline. Specifically, given the multi-view images, 2D backbone and neck (e.g. Swin-tiny[9] and FPN[10]) are used to obtain multi-scale perspective features $P = \{P_l \in R^{N_c \times C_p \times H_p \times W_p}, l = 1, ..., L\}$, where N_c, C_p, H_p and W_p represent the camera number, feature dimension and the perspective feature size. For Lidar points, the irregular and unordered points are first discretized into sparse and ordered voxel features[22] and further encoded with sparse encoder[11] into high-dimensional LiDAR BEV features $B \in R^{C_b \times H_b \times W_b}$, where C_b, H_b, W_b represent the feature dimension and the BEV feature size. Then the view transformation takes the LiDAR BEV features as semantic query encoding and the image perspective features as key/value and output image BEV features in the following module.

B. Optimized View Transformation

The substantial disparities in perspective view features from images and BEV features from LiDAR points require the view transformation to incorporate these multi-modal features into the unified BEV representation. Current state-of-the-art methods either use LSS[7] to lift dense image features into 3D coordinate with predicted depth [2], [3] or leverage cascaded transformer and 3D reference points to fetch sparse image features[4], [36]. In this section, we delve into the mechanism of the two mainstream view transformations and find their common ground that enables feature transferring as well as different emphases that lead to different performance boosts. With the combination of LSS-based and transformer-based methods, the proposed LSSAttn capitalizes on the LSS to establish dense associations and the attention for feature aggregation.

1) Multi-scale Feature Association: After sparse encoding, the points are further partitioned and reshaped into LiDAR BEV features with N non-empty features, and the M multi-scale image features can also be accessed as follows:

$$B = \{b_i | b_i = [f_i^b; d_i^b; (x_i, y_i)]\}_{i=1}^{N} \tag{1}$$

$$P = \{p_i | p_i = [f_i^p; d_i^p; (u_i, v_i)]\}_{i=1}^{M} \tag{2}$$

where $f_i^b \in R^{C_b}$ denotes the LiDAR BEV features and $d_i^b \in [0, H_b \times W_b]$ are the corresponding BEV indexes. $f_i^p \in R^{C_b}$ denotes the perspective features, $d_i^p \in [0, \sum_{l=0}^{L} H_p^l \times W_p^l]$ is the corresponding perspective indexes, L is the scale number and H_p^l, W_p^l is the size of l_{th} perspective features. Following other LSS-based methods, we establish dense associations between the BEV grids and the multi-scale perspective pixels as in Fig.3:

$$Pairs(d^b, d^p) = LSS(d^b, d^p, Calib) \tag{3}$$

where $Calib$ are the calibration parameters and $Pairs \in R^{K \times 2}$ are the K association pairs, including intrinsic and extrinsic parameters. The association pairs are fixed after proper calibration as suggested in BEVFusion[2] and can be pre-computed and cached for fast access. During inference, the pre-computed associations are loaded and masked with real-time valid LiDAR BEV features, where the voxels are not empty.

2) Attention-based Feature Aggregation: LSS has mapped the image features along the ray direction with calibration parameters and redundantly associated the perspective pixels with BEV grids. Then we use attention to calculate the weights and accurately aggregate the corresponding features.

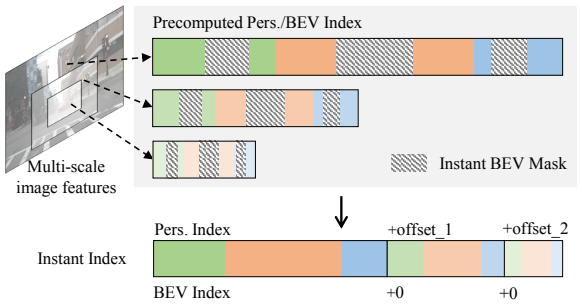


Fig. 3. **Multi-scale feature association.** Fast association with pre-computation is applied in each scale of image features. The instant BEV mask is utilized to reduce the unnecessary attention pairs and the multi-scale associations are concatenated with added offsets.

For redundancy needs in occasions like partial occlusion, objects far away, or with sparse reflected points, grid center position encoding can be used to calculate the geometric similarity in case of sub-optimal LiDAR BEV features or camera-only methods. We use MLP to take the grid center position as input to generate geometric positional encoding[27].

Shifted Subset Partition. LSS[35] densely associates image pixels with BEV grids so the supervision signals back to the perspective features are still dense. However, such dense associations complicate the attention-based similarity calculation. On the one hand, unlike vanilla attention[37] where queries interact with all the keys and values, each query is associated, hence interacts, with a certain amount of keys/values in our cases. On the other hand, BEV grids are associated with an uneven amount of image features due to the perspective principle, varying from dozens to thousands, unfriendly for parallel computation.

To address the aforementioned problems, we propose to conduct attention within the associated subsets and repetitively access subset features instead of mapping them into the vanilla attention[37] format. Specifically, each perspective pixel is associated with a comparative amount of BEV grids as in Fig.4-(a), hence regarded as a subset for weight calculation. Likewise, each BEV grid is associated with multiple perspective pixels and can be regarded as a subset for feature aggregation as in Fig.4-(b). For each perspective pixel, the associated BEV indexes can be obtained:

$$S_i^b = Pairs[d^p == d_i^p] \tag{4}$$

where S_i^b is the BEV index subsets associated with the i_{th} perspective index d_i^p. Likewise, we can obtain the perspective index subsets S_i^p associated with the i_{th} BEV index d_i^b.

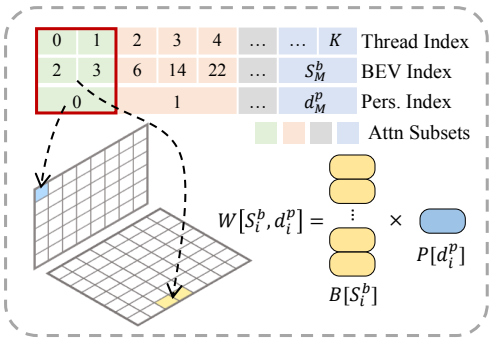
Feature Aggregation within Subsets. The associated perspective and BEV features are accessed with the index for scale-dot attention[37] and we apply the softmax function in the subsets where one perspective pixel is associated with multiple BEV grids:

$$W[S_i^b, d_i^p] = softmax(\frac{B[S_i^b] \cdot (f_i^p)^T}{\sqrt{d_k}}) \tag{5}$$

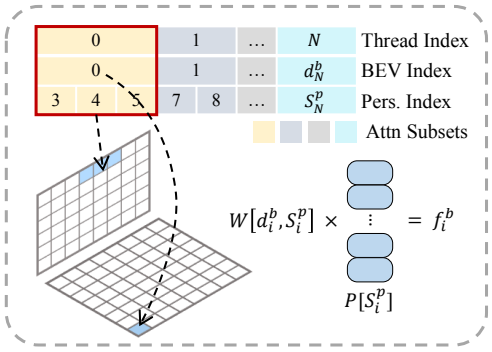
where $W = \{W_i | W_i = [w_i; d_i^p; d_i^b]\}_{i=1}^K$ are the calculated scale-dot attention weights and can be accessed with d^p and d^b, w_i is the calculated weight. Each image feature is associated with a relatively stable number of LiDAR BEV features even after the BEV masking. The weights calculated in this way have a similar function to the predicted depth in LSS while transformer-based attention weights focus on the most relevant perspective features among all. Then we calculate the feature aggregation in the shifted subsets, where one BEV feature is associated with multiple perspective features:

$$f_i^b = W[d_i^b, S_i^p] \cdot P[S_i^p] \tag{6}$$

Optimized Acceleration Kernel. Instead of padding the LiDAR BEV features and camera perspective features into the standard cross-attention, we implement optimized GPU kernels that parallel the attention weight calculation and weighted feature aggregation process. We assign a GPU thread to calculate the attention weights and feature aggregation. This kernel uses indexes within the subsets to access features and avoid extra memory consumption of repetitive elements and stores weights for latter feature aggregation. Then, another kernel is utilized to aggregate image features with previous weights for each image BEV feature.



(a) Similarity-based weight calculation



(b) Feature aggregation for BEV feature

Fig. 4. **Attention-based Feature Aggregation.** Multi-scales and multi-heads are omitted for simplicity. The association pairs are first split into subsets with one perspective feature and multiple LiDAR BEV features in (a) to calculate similarity as weights. Then subsets with one BEV feature and multiple perspective features are constructed in (b) for feature aggregation.

TABLE I

COMPARISONS WITH THE STATE-OF-THE-ART BEV-BASED 3D DETECTION METHODS ON NUSCENES VALIDATION SET. THE PROPOSED VT MODULE EFFECTIVELY AND EFFICIENTLY BOOSTS THE PERFORMANCE OF THE LSS-BASED BASELINE MODELS, INCLUDING CAMERA-ONLY AND MULTI-MODAL BASELINE[2]. (†: WITH TEMPORAL SEQUENCES. ‡: SINGLE-SCALE IMAGE FEATURE AS THE BASELINE)

Method	Modality	2D Backbone	Resolution	mAP	NDS	VT Params(M)	Params(M)
TransFusion[27]	L	-	-	64.2	69.2	-	8.3
BEVFormer-S[6]	C	Res101	900×1600	40.9	46.2	3.6	67.8
BEVFormer†[6]	C	Res101	900×1600	53.5	44.5	3.6	69.1
Fast-BEV[14]	C	Res101	900×1600	40.2	53.1	-	-
BEVDet[8]	C	Swin-T	256x704	29.4	38.4	-	53.7
BEVFusion-MIT-C[2]	C	Swin-T	256×704	35.6	41.2	1.2	44.3
+LSSAttn(Ours)‡	C	Swin-T	256×704	36.3	42.2	20.1	63.4
FUTR3D[4]	L+C	Res101	900×1600	64.2	68.0	8.5	66.6
BEVFusion-PU[3]	L+C	Dual-Swin-T	448×800	69.6	72.1	16.4	90.3
BEVFusion4D-S[36]	L+C	Dual-Swin-T	448×800	70.9	72.9	-	71.7
EA-LSS[35]	L+C	Swin-T	256×704	69.4	71.8	-	-
BEVFusion-MIT-LC[2]	L+C	Swin-T	256×704	68.5	71.4	2.6	40.8
+LSSAttn(Ours)	L+C	Swin-T	256×704	69.8	71.7	8.7	47.4

TABLE II

ABLATION STUDY OF THE DIFFERENT VIEW TRANSFORMATIONS. ALL THE VT MODULES TAKE THE SINGLE-SCALE CAMERA PERSPECTIVE FEATURES AND LIDAR BEV FEATURES AS INPUT AND OUTPUT IMAGE BEV FEATURES WITH THE SIZE OF 180×180. (‡: TRANSFUSION-L AS THE BASELINE, SAME BASELINE USED IN BEVFUSION-MIT[2]. †: ADAPTED FROM THE SPATIAL CROSS-ATTENTION FROM BEVFORMER[6] WITH LIDAR BEV FEATURES AS EXTRA SEMANTIC POSITIONAL ENCODING, 3 LAYERS. §: TRAINING MEMORY REQUIREMENT.)

View Trans.	mAP↑	NDS↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓	Memory(Mb)§	Params(M)	FPS
LiDAR Only‡	64.2	69.2	29.0	**25.3**	**29.5**	26.3	19.2	18125	8.3	6.7
LSS-uniform	68.3	71.1	**28.9**	25.5	31.6	26.3	18.6	23707	40.4	4.6
LSS-predict	68.6	71.3	29.0	25.6	31.1	**26.0**	18.9	23793	40.5	4.6
Deform Attn†	68.9	71.2	29.0	25.4	33.0	26.7	18.6	24105	47.4	4.7
Proposed	**69.5**	**71.8**	29.2	25.5	29.7	26.6	**18.4**	21777	49.8	4.5

C. Fusion Module and Detection Head

With the image perspective features transferred into the unified BEV representation as its LiDAR counterpart, we adapt convolutional fuser and BEV decoder for fusion following BEVFusion[2], [3]. TransFusion[27] head is applied with class-specific center heatmap prediction for query initialization and 3D attributes regression heads, including dimensions, offsets to grid center, heading angle, and velocity. Please refer to TransFusion[27] for more details about the detection head.

IV. EXPERIMENTS

A. Dataset and Evaluation Metrics

We validate the proposed model on nuScenes dataset, one of the largest multimodal autonomous-driving datasets. For the comprehensive evaluation of the 3D detection task, we follow the official evaluation metrics including mean Average Precision (mAP) and nuScenes detection score (NDS).

Implementation Details: As a plug-and-play module, LSSAttn can be easily modulated to LSS-based models. In our experiments, we use BEVFusion-MIT-LC[2] and BEVFusion-MIT-C[2] as our baseline models to verify the effectiveness of the proposed view transformation. The training settings are the same as the baseline except that we set the frozen stage of the image backbone to 3 to save the memory during training. We conduct our experiments on 4 Nvidia GTX 3090Ti GPUs with a batch size of 4 and the learning rate and warm-up iterations will be linearly adjusted to the total batch size.

B. Comparison with the SOTA methods

The experimental result comparisons on the nuScenes validation with the state-of-the-art methods in Table I. Compared to the baseline model, the proposed module increases 1.3% in mAP and 0.3% in NDS. Our proposed module can be extended to camera-only LSS-based models as well due to the geometric positional encoding and it achieves a 0.7% increase in mAP and 1.0% increase in NDS even with single-scale features as the baseline.

C. Ablation study

In this section, we conduct the ablation study on some important modules of the proposed LSSAttn. All the experiments are conducted using pre-trained Swin-tiny[9] as the backbone and a single-level image feature with the stride of 8 unless explicitly specified.

View Transformation. We choose BEVFusion-MIT[2] as the baseline and replace the vanilla view transformation with other view transformation modules for fair comparisons in

TABLE III

ABLATION STUDY OF THE VIEW TRANSFORMATION DESIGN.

modules	pos. enc.	feat. stride	value embed	q/k proj.	v/out proj.	mAP	ΔmAP
Q/K/V Initialization	lidar & mlp	8, 16	mlp for score	non-linear	none	69.8	
	lidar & mlp	8, 16	mlp for score	linear	none	69.1	-0.7
	lidar & mlp	8, 16	mlp for score	non-linear	linear	68.8	-1.0
	lidar & mlp	8, 16	learned	non-linear	none	69.1	-0.7
Position Encoding	lidar & mlp	8	mlp for score	non-linear	none	69.3	
	lidar	8	mlp for score	non-linear	none	69.1	-0.2
Multiply Features	lidar & mlp	8, 16	mlp for score	non-linear	none	69.8	
	lidar & mlp	4	mlp for score	non-linear	none	68.3	-1.5
	lidar & mlp	8	mlp for score	non-linear	none	69.5	-0.3
	lidar & mlp	4, 8, 16	mlp for score	non-linear	none	69.0	-0.8

Table II. The uniform distributed LSS significantly improves the performance of the LiDAR-only method by a 4.1% increase in mAP and a 1.9% increase in NDS. The LSS method with predicted depth distribution are slightly superior with an increase of 0.3% in mAP and 0.2% in NDS than uniformly distributed LSS. This indicates that instead of retrieving the geometric structure of objects with pseudo points, the dense pseudo points guarantee the correct but redundant mapping of image features to BEV grids. The predicted depths are used as weights to aggregate features in BEV pooling and also adjust the gradients backward to perspective features. We also compare the 3-layer spatial cross-attention adapted from BEVFomer[6] to fetch image features. The spatial cross-attention multiplies the feature association with the cascaded layers, though still sparser than LSS-based methods. It achieves comparable performance to the LSS-based method with slightly more memory consumption. Our proposed methods combine the dense feature association and attention-based feature aggregation and achieve an increase of 5.3% in mAP and 2.6% in NDS compared to the LiDAR-only baseline. Compared with the LSS method with predicted weights which uses the same feature association but different weight calculation, LSSAttn achieves 0.9% in mAP and 0.5% in NDS, indicating the effectiveness of the attention-based feature aggregation mechanism.

Q/K/V Initialization. We use dense queries with the same size as LiDAR BEV features. The queries are initialized from learnable parameters while key and value are initialized from image features. The LiDAR features serve as the query positional encoding to calculate the semantic similarity. Apart from the semantic similarity, we also use grid center information for extra positional similarity in case of sub-optimal LiDAR features and camera-only methods. Methods with transformers in single modality usually use linear projection for query and key. Instead, we use a 2-layer MLP for non-linear projection, following cross-modality contrastive learning[38]. Also, to minimize the disparity between image perspective features and image BEV features, we eliminate the projection for value and output. These design choices are ablated in Table III. The linear projection for query and key and value/output projection brings along a decrease of 0.7% and 1.0% in mAP separately. We also compare the multi-scale embedding for value between learnable embedding[6] and MLP to generate scores of each pixel feature for multi-scale features. The learnable embedding is 0.7% inferior in mAP.

Positional Encoding. The positional encoding slightly contributes to the detection performance with a 0.2% increase in mAP. This may happen when BEV feature and position encoding are added and the BEV features are the dominant part when calculating similarity. It indicates the superiority of semantic information in multi-modality data.

Multi-scale Image Features. The image feature with the stride of 4 consumes more memory but brings no performance boost. On the other hand, the image feature with the stride of 32 is too small (8×11 in this experiment) so the pseudo points are too sparse to correct associate BEV grids and perspective pixels. So we mainly utilize image features with the stride of 8 and 16, and the introduction of an extra feature map with the stride of 16 brings a 0.3% increase in mAP with slightly extra memory consumption.

Optimized Kernels and Non-optimized Kernels. The comparisons in Fig.I-(b) are conducted with key/value size as 256×704 and feature channel as 256, query size as 180×180 and feature channel as 256, depth interval as 0.6m, multi-head number as 8. We clip the association number to 63, otherwise the length of the value is too large to be processed. To pad the features for vanilla cross-attention computation[37], the non-optimized kernels, gather and scatter are used, which takes approximately 4.7s and 9807Mb. With our optimized kernels, it highly speeds up the attention within the subsets into 8ms and avoids the repetitive elements in padding.

V. CONCLUSIONS

In this paper, we delve into the view transformation in multi-modal BEV perception and decompose the functional essences of the LSS-based and transformer-based view transformation into feature association and feature aggregation. We propose an optimized view transformation, namely LSSAttn, to combine the dense association of LSS and accurate aggregation of the attention mechanism. The optimized kernels enable the cross-attention within subsets. The proposed optimized view transformation achieves superior detection performance than other VT modules and boosts the performance of BEVFusion-MIT[2].

REFERENCES

[1] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. *arXiv preprint arXiv:1903.11027*, 2019.

[2] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. *arXiv preprint arXiv:2205.13542*, 2022.

[3] Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. BEVFusion: A Simple and Robust LiDAR-Camera Fusion Framework. *arxiv*, 2022.

[4] Xuanyao Chen, Tianyuan Zhang, Yue Wang, Yilun Wang, and Hang Zhao. Futr3d: A unified sensor fusion framework for 3d detection. *arXiv preprint arXiv:2203.10642*, 2022.

[5] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. *arXiv preprint arXiv:2206.10092*, 2022.

[6] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. *arXiv preprint arXiv:2203.17270*, 2022.

[7] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *European Conference on Computer Vision*, pages 194–210. Springer, 2020.

[8] Junjie Huang, Guan Huang, Zheng Zhu, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021.

[9] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021.

[10] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection, 2017.

[11] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018.

[12] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6569–6578, 2019.

[13] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 913–922, 2021.

[14] Yangguang Li, Bin Huang, Zeren Chen, Yufeng Cui, Feng Liang, Mingzhu Shen, Fenggang Liu, Enze Xie, Lu Sheng, Wanli Ouyang, et al. Fast-bev: A fast and strong bird's-eye view perception baseline. *arXiv preprint arXiv:2301.12511*, 2023.

[15] Adam W. Harley, Zhaoyuan Fang, Jie Li, Rares Ambrus, and Katerina Fragkiadaki. Simple-BEV: What really matters for multi-sensor bev perception? In *arXiv:2206.07959*, 2022.

[16] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, pages 180–191. PMLR, 2022.

[17] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. *arXiv preprint arXiv:2203.05625*, 2022.

[18] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[19] Xuran Pan, Zhuofan Xia, Shiji Song, Li Erran Li, and Gao Huang. 3d object detection with pointformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7463–7472, June 2021.

[20] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[21] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12697–12705, 2019.

[22] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4490–4499, 2018.

[23] Haiyang Wang, Chen Shi, Shaoshuai Shi, Meng Lei, Sen Wang, Di He, Bernt Schiele, and Liwei Wang. Dsvt: Dynamic sparse voxel transformer with rotated sets. In *CVPR*, 2023.

[24] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[25] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[26] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021.

[27] Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1090–1099, 2022.

[28] Yukang Chen, Jianhui Liu, Xiangyu Zhang, Xiaojuan Qi, and Jiaya Jia. Voxelnext: Fully sparse voxelnet for 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[29] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1907–1915, 2017.

[30] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 918–927, 2018.

[31] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. Pointaugmenting: Cross-modal augmentation for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11794–11803, 2021.

[32] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4604–4612, 2020.

[33] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Multimodal virtual point 3d detection. *Advances in Neural Information Processing Systems*, 34:16494–16507, 2021.

[34] Qi Jiang, Hao Sun, and Xi Zhang. Semanticbevfusion: Rethink lidar-camera fusion in unified bird's-eye view representation for 3d object detection, 2022.

[35] Haotian Hu, Fanyi Wang, Jingwen Su, Yaonong Wang, Laifeng Hu, Weiye Fang, Jingwei Xu, and Zhiwang Zhang. Ea-lss: Edge-aware lift-splat-shot framework for 3d bev object detection. *arXiv preprint arXiv:2303.17895*, 2023.

[36] Hongxiang Cai, Zeyuan Zhang, Zhenyu Zhou, Ziyin Li, Wenbo Ding, and Jiuhua Zhao. Bevfusion4d: Learning lidar-camera fusion under bird's-eye-view via cross-modality guidance and temporal aggregation, 2023.

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Adv. Neural Inf. Proces. Syst.*, 30, 2017.

[38] Zhenyu Li, Zehui Chen, Ang Li, Liangji Fang, Qinhong Jiang, Xianming Liu, Junjun Jiang, Bolei Zhou, and Hang Zhao. Simipu: Simple 2d image and 3d point cloud unsupervised pre-training for spatial-aware visual representations. *arXiv preprint arXiv:2112.04680*, 2021.