UniBEV: Multi-modal 3D Object Detection with Uniform BEV Encoders for Robustness against Missing Sensor Modalities

Shiming Wang, Holger Caesar, Liangliang Nan, Julian F. P. Kooij

***Abstract*— Multi-sensor object detection is an active research topic in automated driving, but the robustness of such detection models against missing sensor input (modality missing), e.g., due to a sudden sensor failure, is a critical problem which remains under-studied. In this work, we propose UniBEV, an end-to-end multi-modal 3D object detection framework designed for robustness against missing modalities: UniBEV can operate on LiDAR plus camera input, but also on LiDAR-only or camera-only input without retraining. To facilitate its detector head to handle different input combinations, UniBEV aims to create well-aligned Bird's Eye View (BEV) feature maps from each available modality. Unlike prior BEV-based multi-modal detection methods, all sensor modalities follow a uniform approach to resample features from the original sensor coordinate systems to the BEV features. We furthermore investigate the robustness of various fusion strategies w.r.t. missing modalities: the commonly used feature concatenation, but also channel-wise averaging, and a generalization to weighted averaging termed Channel Normalized Weights. To validate its effectiveness, we compare UniBEV to state-of-the-art BEVFusion and MetaBEV on nuScenes over all sensor input combinations. In this setting, UniBEV achieves better performance than these baselines for all input combinations. An ablation study shows the robustness benefits of fusing by weighted averaging over regular concatenation, and of sharing queries between the BEV encoders of each modality. Our code will be released upon paper acceptance.**

I. INTRODUCTION

THE perception system of an intelligent vehicle typically relies on multiple sensors [1], [2], including LiDARs and cameras, to take advantage of their individual strengths and complementary nature for robust object detection. For example, cameras provide rich texture information while LiDAR delivers dense point clouds with accurate geometric information. Most works on multi-sensor models focus only on optimal detection performance when all sensors are available. However, ideally a model could also be used when the input of one of its sensors is missing, i.e., modality missing, *without any retraining*. A unified model to handle both multi-sensor and single-sensor inputs would facilitate gracefully degradation of its perception system in case of catastrophic sensor failure (e.g., broken connector) without needing to load a new set of model parameters, but also provide flexibility by supporting diverse hardware configurations (e.g., vehicles with different sensors). This work therefore focuses on the design of a 'robust' multi-sensor object detection model, which in this context refers to the trained model's ability to fuse camera and LiDAR information for object detection, but also to operate on only a single modality.

All authors are with TU Delft, Delft, the Netherlands. {s.wang-15, h.caesar, liangliang.nan, j.f.p.kooij}@tudelft.nl

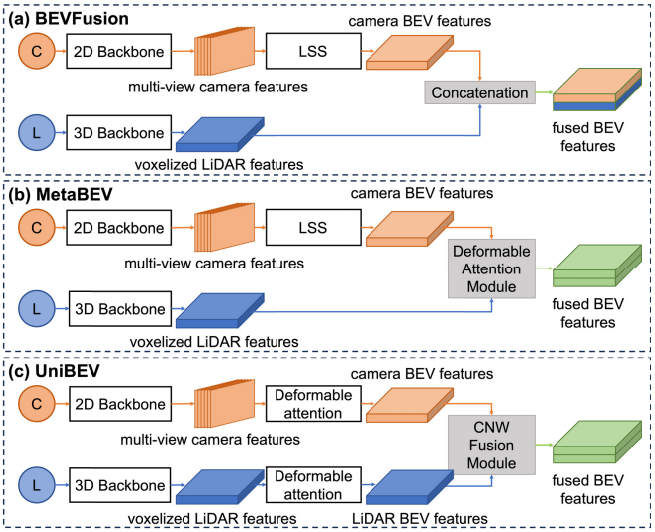

Fig. 1. **Comparison of our UniBEV with other relevant works.** (a). BEVFusion [3] fuses multi-modal BEV features extracted from two separate branches with concatenation. (b) MetaBEV [4] fuses multi-modal BEV features extracted from two separate branches with a fusion module consisting of several deformable attention layers. (c) Our UniBEV extracts multi-modal BEV features from their original coordinate systems with uniform BEV encoders and fuses the BEV features with the CNW module. C and L in the figure represent the input from cameras and LiDAR.

Recent state-of-the-art (SotA) in multi-sensor object detection for self-driving exploits dense Bird's-Eye view (BEV) features as an intermediate representation to integrate multi-sensor information, which can then be used by a generic object detection head. BEVFusion [5] pioneers fusing multi-modal BEV features for LiDAR and cameras. It employs two separate branches to independently extract BEV features from each modality, and subsequently fuses these features through concatenation, as shown in Fig. 1 (a). Notably, the designs of the camera and LiDAR branches in BEV-Fusion are non-uniform: The camera branch relies on a (type of) Lift-Splat-Shoot (LSS) [6] component to explicitly predict the depth distribution of the image features, and map them from their camera coordinates to the spatial BEV coordinates. In contrast, the LiDAR branch already natively expresses its features in spatial coordinates and thus does not apply additional transformations to encode its BEV features. This may lead to the misalignment of camera and LiDAR BEV features as they are extracted in distinct ways. Recently MetaBEV [4] improves over BEVFusion by replac-

ing its concatenation with a learnable module with several deformable attention layers to better align the features, while still keeping BEVFusion's BEV feature encoder approach, see Fig. 1 (b). Since the feature misalignment is not really solved, the gain of the fusion inference is modest.

We argue that for 'robust' multi-modal 3D object detection, i.e., for both multi-sensor and single-sensor inputs without retraining, it is important that the BEV representations of all sensor modalities are well aligned. We therefore propose a new end-to-end model named *UniBEV*, shown in Fig. 1 (c), which revisits several key architectural design choices to improve feature alignment: First, it uses a uniform deformable attention-based architecture for both its camera and LiDAR branch to build each sensor's BEV features, avoiding the need for a camera-only LSS-like explicit depth prediction. Now both branches use deformable attention to construct their BEV features, and the learned queries can be shared between both branches to further facilitate feature alignment and provide interactions between the two branches. Second, to fuse the multi-sensor features, we investigate using simple averaging over concatenation to avoid zeroing-out half the features when only a single sensor is available. We also propose an extension to a learned weighted average of feature channels, called Channel Normalized Weights (CNW).

Our main contributions are as follows:

- We propose UniBEV, a multi-modal 3D object detector designed for robustness against missing modalities. It follows a uniform approach across all modalities to encode the sensor-specific features into a shared BEV feature space to facilitate alignment between modalities. UniBEV exhibits a more robust performance than its baselines against modality missing failure without needing to load a different set of model parameters.
- When taking multi-modal data as input, UniBEV outperforms SotA multi-modal methods on the nuScenes dataset. For fair benchmarking, we reimplemented the closed-source MetaBEV and evaluated all multi-modal baselines under similar training conditions (e.g. same hardware, no data augmentation). We will release all source code upon paper acceptance.
- We investigate the impact of various feature fusion strategies: concatenation, averaging, and a simple extension to weighted averaging we call Channel Normalized Weights. For the same number of feature channels after fusion, the CNW performs better when modality missing is considered than the commonly used fusion by feature concatenation.

II. RELATED WORK

Every sensor type has specific limitations for real-world driving scenarios, hence multi-modal 3D object detection has gained great attention in recent years, especially the fusion between cameras and LiDAR. Nevertheless, the alignment between camera and LiDAR features is challenging as they are defined in different coordinates. Some methods fuse multi-modal features directly from their native coordinates with some specially designed components, such as attention [7]–[11]. DeepFusion [12] is the representative work among them, it simply performs cross-attention on multi-modal features with LiDAR features as queries and with camera features as keys and values. FUTR3D [13] brings the idea of DETR3D [14] and Object DGCNN [15] into the multi-modal domain. With the prior knowledge of camera extrinsics, FUTR3D leverages the shared object queries interacting with multi-view image features and LiDAR BEV features to sample instance-level features.

Other SotA methods build a unified intermediate BEV representation to align and fuse multi-modal features [16], [17]. A benefit of such BEV representations is they can serve various tasks through distinct network heads, such as simultaneous object detection and BEV map segmentation [5], [6], [18]–[20]. BEVFusion [3], [5] deploys Lift-Splat-Shoot (LSS) [6] to predict image depth distributions and project the image features in BEV, and deploys a regular point voxelization method, such as PointPillars [21] or CenterPoint [22], to extract BEV features from LiDAR point clouds. The multi-modal BEV feature maps are fused by concatenation. MetaBEV [4] upgrades the fusion module of BEVFusion to a deformable attention-based fusion block. [23] uses a simple summation fusion module to integrate cross-modal BEV features and achieve convincing performance at a long distance. In this study, our primary emphasis is on the alignment within the BEV feature domain to address the challenges posed by missing sensor scenarios.

A few works have looked into increasing robustness against modality missing [4], [24], [25]. These improve robustness by applying *Modality Dropout* during training, i.e., some portions of the training samples are presented without the input of one of the sensors. In this paper, we focus on this Modality Dropout setting, and explore a model's holistic test performance when presented with both or only one input modality.

III. METHODOLOGY

We now describe our new architecture, UniBEV, for robust LiDAR-camera 3D object detection. As illustrated in Fig. 2, UniBEV consists of four parts: features extractors, uniform BEV encoders, a fusion module, and the detection head. Each part will be described in the following subsections.

A. Feature Extractors

For the initial feature extraction, UniBEV has a similar design to previous works [14], [18], relying on common image/point cloud backbones [3], [5], [13], [25]. Images from V camera views are fed into an image backbone, such as ResNet [26], resulting in image features $F_{\mathbf{C}}^{i} \in \mathbb{R}^{H_{\mathbf{C}} \times W_{\mathbf{C}} \times N_{\mathbf{C}}}$ for $1 \leq i \leq V$, where $H_{\mathbf{C}} \times W_{\mathbf{C}}$ is the resolution of the feature map in the native image coordinates, and $N_{\mathbf{C}}$ is the feature dimension. Similarly, the LiDAR scan is processed by a regular point cloud backbone, such as VoxelNet [27], which voxelizes and extracts grid-shaped features in Bird's Eye view $F_{\mathbf{L}} \in \mathbb{R}^{H_{\mathbf{L}} \times W_{\mathbf{L}} \times N_{\mathbf{L}}}$, where $H_{\mathbf{L}}$, $W_{\mathbf{L}}$, and $N_{\mathbf{L}}$ are the spatial shape and feature dimensions of the features.

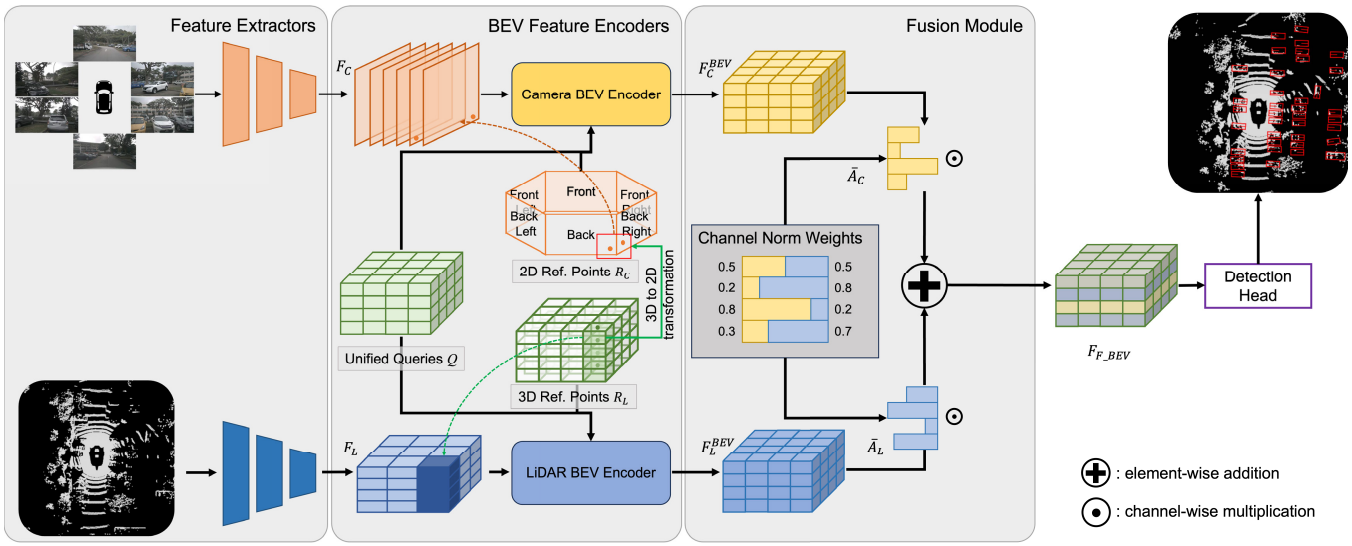

Fig. 2. **The overall architecture of the UniBEV framework.** 1). Multi-view images and point clouds are processed through their respective backbones to generate multi-modal features. 2). A predefined set of grid-shaped BEV queries, shared across modalities, is utilized. Guided by these shared BEV queries, modality-specific BEV encoders further refine the camera and LiDAR features independently to establish aligned BEV features. These encoders are constructed using deformable attention modules and accept unified queries, relevant reference points, and the backbone-extracted features as inputs. 3). The camera and LiDAR BEV features are fused along the channels according to the learned CNW values.

B. Uniform BEV Feature Encoders

After feature extraction, $F_{\mathbf{L}}$ and $F_{\mathbf{C}}$ are still represented in different coordinate systems. $F_{\mathbf{L}}$ is expressed in 3D spatial coordinates similar to the target BEV space, while $F_{\mathbf{C}}$ uses 2D image coordinates. Existing methods generally further transfer the image features into the Bird's Eye view with LSS [6] and simply fuse the two BEV features through concatenation [5], [28]. We argue that the difference in network architecture between these branches may affect the alignment of the camera and LiDAR BEV features. Furthermore, concatenating features requires zero-filling when one modality is missing. As a result, the decoder head would operate on BEV features that are highly different depending on the available inputs, which may impact its robustness to a missing modality. UniBEV therefore implements a uniform design for all sensor modalities for better aligned BEV features, as explained next.

Queries: First, a set of learnable BEV query vectors [18] with associated 3D spatial locations is defined. These queries are shared by all modalities (our ablation study will also consider separate queries per modality). We define learnable parameters $Q \in \mathbb{R}^{H \times W \times N}$ as BEV queries, where $H \times W$ represent the 2D BEV spatial grid resolution in the vehicle's local spatial coordinates, and N is the number of channels in the BEV queries. $R \in \mathbb{R}^{D \times H \times W \times 4}$ contains the corresponding spatial coordinates of BEV reference points in a 3D spatial grid $H \times W \times D$ as homogeneous coordinates $(x, y, z, 1)$. Note that D reference locations are defined along the z-direction in the pillar of each 2D query location in Q. We shall use $R(z)$ to denote only the references at level $1 \leq z \leq D$.

Projection: The BEV spatial locations R are projected to the original spatial coordinate system of each modality's feature map, as shown in Fig. 2, similar to FUTR3D [13] [1]. Namely, for the feature map of each camera $i = \{1, 2, ..., V\}$, the 3D points R are projected to its 2D image-based coordinates $R^i_{\mathbf{C}} = \mathsf{P}_{\mathbf{C}}(R, P^i)$ using the known camera extrinsics expressed by its homogeneous projection matrix P^i. Similarly, $R_{\mathbf{L}} = \mathsf{P}_{\mathbf{L}}(R)$ projects the references to the LiDAR feature map's spatial coordinates, for instance to scale the spatial resolution, though in practice often $\mathsf{P}_{\mathbf{L}}$ is an identity function.

Encoding: Finally, each modality's BEV feature map is constructed using 3 layers of deformable self-attention and deformable cross-attention between the BEV queries and sensor feature maps. The feature map at the first layer of the camera BEV encoder, $F^{BEV'}_{\mathbf{C}}$, is obtained by summing over all views where a reference is visible, and over all D locations for each query [18],

$$F^{BEV'}_{\mathbf{C}} = \sum_{1 \leq i \leq V} \sum_{1 \leq z \leq D} DeformAttn(Q, R^i_{\mathbf{C}}(z), F^i_{\mathbf{C}}), \quad (1)$$

where $DeformAttn$ is the deformable cross-attention defined in [29], [30]. The output of the last layer is the final camera BEV feature map $F^{BEV}_{\mathbf{C}}$ passed to the fusion module.

Mirroring the cameras, the LiDAR BEV encoder performs the same operations, with its first feature map likewise

$$F^{BEV'}_{\mathbf{L}} = \sum_{1 \leq z \leq D} DeformAttn(Q, R_{\mathbf{L}}(z), F_{\mathbf{L}}). \quad (2)$$

Note that due to how deformable attention works, both $F^{BEV}_{\mathbf{C}}$ and $F^{BEV}_{\mathbf{L}}$ will retain the $H \times W \times N$ size of the initial Q.

C. Fusion Module: Channel Normalized Weights

In the majority of multi-modal 3D object detection approaches [3], [5], [13], [25], [28], *concatenation* along the

[1] Recall FUTR3D is not a BEV-based detector, nor does it use deformable attention to sample camera features.

feature dimension is utilized as the fusion method to combine features from different modalities, to preserve a maximum amount of information. However, if one considers scenarios where a sensor input is missing, concatenation fusion must compensate for the absent input to ensure the number of channels provided to the decoder does not change. For instance, by filling the fused BEV features with placeholder values, typically all zeros.

We shall investigate a simple alternative, namely fusing BEV feature maps by *averaging* (or summing [23]) over all available modality feature maps. On the one hand, averaging risks diluting information from a more reliable sensor with that of the less reliable sensor. On the other hand, this fusion strategy never needs to resort to placeholder values, and ensures that the fused BEV feature map always has the same number of channels as each modality BEV feature map, even if one input modality is missing.

We also propose a generalization of average fusion which we call *Channel Normalized Weights (CNW)*. The key idea is that the reliability of each channel in the feature map may differ from sensor to sensor, and we should account for this when sensor measurements can be fused. CNW therefore has as learnable parameters a N-dimensional vector A_m for each modality m, which remains fixed after training. The i-th element $A_m(i)$ indicates the relative importance of modality m for the fused result of the i-th channel. Before fusion, weights A_m are normalized (denoted $\overline{A}_m$) over all available sensor modalities such that they sum to 1 per channel. Thus with two modalities, LiDAR and camera, $\overline{A}_{\mathbf{C}}(i), \overline{A}_{\mathbf{L}}(i) = softmax(A_{\mathbf{C}}(i), A_{\mathbf{L}}(i))$, s.t.

$$CNW(F_{\mathbf{C}}^{BEV}, F_{\mathbf{L}}^{BEV}) = F_{\mathbf{C}}^{BEV} \odot \overline{A}_{\mathbf{C}} + F_{\mathbf{L}}^{BEV} \odot \overline{A}_{\mathbf{L}}, \quad (3)$$

where $\odot$ indicates channel-wise multiplication with implied broadcasting over the spatial dimensions. In case only a single modality is available, $softmax$ is applied to a single value per channel and normalization reduces to assigning full weights to that modality, e.g., $CNW(F_{\mathbf{C}}^{BEV}) = F_{\mathbf{C}}^{BEV}$.

It is easy to see that CNW reduces to average fusion when all the learned channel weights in $\overline{A}_{\mathbf{C}}$ and $\overline{A}_{\mathbf{L}}$ approach $\frac{1}{2}$. On the other hand, CNW can also reflect concatenation fusion by allowing channels in the fused output to only take information from one modality if those channels' learned weights approach 0 or 1 only. Intuitively, CNW adds a small number of learnable parameters to give the model more flexibility between these special cases, allowing it to optimize the relative importance of each modality for fusion, and still allowing meaningful values for the single modal input. Our experimental results shall show UniBEV constructs BEV features with a similar magnitude distribution for each modality, ensuring that our CNW discerns the importance of different channels rather than a random scale function.

D. Detection Head and Modality Dropout Strategy

Following previous works [13], [14], [18], [31], we cast the bounding box detection as a set prediction problem and adopt the decoder of BEVFormer [18] for 3D object detection task. To train the model for sensor missing failure, we deploy the common *Modality Dropout* (MD) training strategy [4], [24], [25]. Thus during training we drop with a probability p_{md} the BEV features of one of the modalities, either $F_{\mathbf{C}}^{BEV}$ or $F_{\mathbf{L}}^{BEV}$. Furthermore, in case we do drop one of the modalities, p_L indicates the probability of keeping LiDAR, while $p_C = 1 - p_L$ is the probability of keeping the cameras. Thus, the overall probability of keeping both sensors is $1 - p_{md}$, of only LiDAR is $p_{md} \cdot p_L$, and of only cameras is $p_{md} \cdot p_C = p_{md} \cdot (1 - p_L)$.

IV. EXPERIMENTS

A. Implementation Details

Dataset and Metrics. We train and evaluate our approach on the nuScenes dataset [2]. nuScences is a large-scale multi-modal driving dataset, which includes 6 cameras and a 32-beam LiDAR in the sensor suite. As we target robustness against missing input modalities, we report the test performance on LiDAR+camera, on LiDAR-only, and on camera-only, and report common mean Average Precision (mAP) and nuScenes detection score (NDS) [2]. As a *summary metric* for 'robustness to modality missing', we also report for both metrics the average performance over all these three possible sensor inputs. For example, our *summary mAP* is simply,

$$summary\ mAP = \frac{1}{3}(mAP_{L+C} + mAP_L + mAP_C). \quad (4)$$

Model. We use ResNet-101 [26] with FPN [32] as UniBEV's camera feature extractor and VoxelNet [27] as its LiDAR feature extractor. CNW is the default fusion approach. The grid shape of the unified query is set to 200×200 with $N = 256$.

Baselines. Our first multi-sensor baseline is BEVFusion [5], and uses their implementation which includes a more powerful image backbone, Dual-Swin-Tiny [33], and CenterPoint [22] head. Since BEVFusion uses concatenation for fusion, we use zero-filling for MD. Our second fusion baseline is the recent MetaBEV [4], which improves the concatenation approach of BEVFusion to a deformable attention-based fusion module. Our MetaBEV implementation uses the same backbones and detector head as UniBEV [2].

To assess if the results on one sensor only are reasonable, we also evaluate related uni-modal baselines: LSS [6], BEVFormer_S [18], PointPillars [21], and CenterPoint [22]. Additionally, we let UniBEV_C denote the camera branch of UniBEV only trained with multi-view images, to compare to the camera-only methods. Likewise, UniBEV_L is the LiDAR-only branch of UniBEV.

Training Details. Our model is trained in an end-to-end manner for 36 epochs. Due to the utilization of various tricks in baseline methods and the absence of publicly available code, accurately reproducing the performance reported in their papers proves to be challenging. For a fair comparison, all the baselines are retrained with the same data pipeline

[2]The code of MetaBEV has not been released yet. We reproduced their BEV-encoder and fusion strategies based on the paper. Their multi-task learning strategy was not implemented for a fair comparison.

TABLE I

EVALUATION RESULTS ON THE NUSCENES *val* SET (**BEST**/ SECOND BEST). COLUMNS INDICATE THE TEST INPUT MODALITIES: L+C = LIDAR AND CAMERAS, L = ONLY LIDAR, AND C = ONLY CAMERAS. NON-FUSION MODELS ARE PROVIDED FOR COMPLETENESS, USING "-" WHERE THEY DO NOT APPLY. NOTE: WE TRAINED ALL MODELS OURSELVES, TO ENSURE EQUAL TRAINING STRATEGIES, AND SINCE [4] HAS NO PUBLIC CODE. THIS LEADS TO LOWER PERFORMANCE COMPARED TO REPORTED BENCHMARK RESULTS [3], [4], ESPECIALLY DUE TO A LACK OF DATA AUGMENTATION.

Method	Train Modality	L + C		L		C		Summary Metric	
		NDS ↑	mAP ↑	NDS ↑	mAP ↑	NDS ↑	mAP ↑	NDS ↑	mAP ↑
LSS [6]	C	-	-	-	-	33.0	28.1	-	-
BEVFormer_S [18]	C	-	-	-	-	**46.2**	**40.9**	-	-
UniBEV_C	C	-	-	-	-	44.3	36.9	-	-
PointPillars [21]	L	-	-	49.1	34.3	-	-	-	-
CenterPoint [22]	L	-	-	**65.4**	57.0	-	-	-	-
UniBEV_L	L	-	-	65.2	**57.8**	-	-	-	-
BEVFusion [3]	L + C (MD)	65.3	58.7	60.6	49.1	29.6	22.6	51.8	43.5
MetaBEV [4]	L + C (MD)	67.5	62.5	65.2	57.8	33.6	25.9	55.4	48.7
UniBEV (ours)	L + C (MD)	**68.5**	**64.2**	**65.3**	**58.2**	**42.4**	**35.0**	**58.7**	**52.5**

as our model without data augmentation techniques, such as CBGS [34]. Every baseline model underwent training on four Nvidia A40 GPUs, with the entire training duration for each model spanning roughly one week.

Unless stated otherwise, for all models the MD probability is $p_{md} = 0.5$, and the probability for keeping each modality is identical, i.e., $p_L = p_C = 0.5$. Therefore, in the whole training process, on average 50% of iterations are trained with multi-modal inputs, 25% with LiDAR-only inputs, and 25% with camera-only inputs.

The image and point cloud backbones are initialized with the weights of FCOS3D [35] and CenterPoint [22], respectively. Our model is implemented in the open-sourced MMDetection3D [36].

B. Multi-Modal 3D Object Detection

Table I showcases the inference performance of the fusion-based detector on both multi-modal input, as well as single-modality input using the same trained weights.

Multi-modal robustness. The summary metrics of our UniBEV (58.7% NDS and 52.5% mAP) significantly surpass the baseline methods, indicating that UniBEV is more robust over varying input modalities. On all input modalities, UniBEV outperforms its multi-modal baselines, achieving 68.5% NDS and 64.2% mAP for LiDAR+camera fusion, especially the difference in camera-only performance is notable. Despite employing a more powerful image backbone [37], BEVFusion lags markedly behind UniBEV when only camera input is available.

Given that the CenterPoint head of BEVFusion and our detection head exhibit comparable detection capabilities (as evidenced by the near identical performance of UniBEV_L and CenterPoint), the performance difference for camera-only between UniBEV and BEVFusion can be attributed to the quality of BEV features and its fusion strategy. Fig. 3 illustrates that compared to BEVFusion, UniBEV's camera and LiDAR BEV features more clearly discern similar object locations and that these are better spatially aligned. Besides, BEVFusion utilizes LSS [6] as the camera BEV encoder to project image features into the BEV feature space. This enforces an inductive bias on its camera BEV features not present in its LiDAR BEV features, as exhibited by the hexagon-shaped outline.

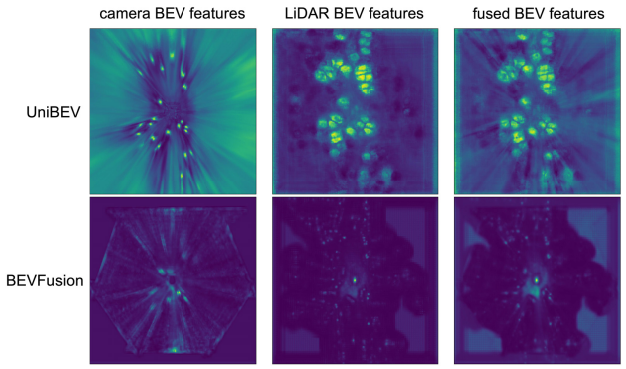

Fig. 3. **Example of the BEV feature maps of different modalities for a single sample.** High intensity indicates high variance across channels at that location. UniBEV aligns modalities for object detection, resulting in strong responses at the same locations in both the camera and LiDAR.

While MetaBEV exceeds BEVFusion across all input types due to its enhanced fusion module, it is also outperformed overall by UniBEV. For the LiDAR-only scenario, MetaBEV does achieve comparable performance to UniBEV, which is unsurprising given the similarities in the LiDAR branch design between UniBEV and MetaBEV. However, similarly to BEVFusion, MetaBEV also adopts LSS as its camera BEV encoder. Although it applies deformable attention to two BEV features instead of our more simple CNW, the BEV feature misalignment cannot be fully compensated for by merely a more powerful fusion strategy.

Qualitative results. To support the comparison between UniBEV and its multi-modal baselines, we show some qualitative detection results in Fig. 4. For instance, we can

see that BEVFusion for camera-only suffers from various false negatives, whereas MetaBEV tends to have more false positives.

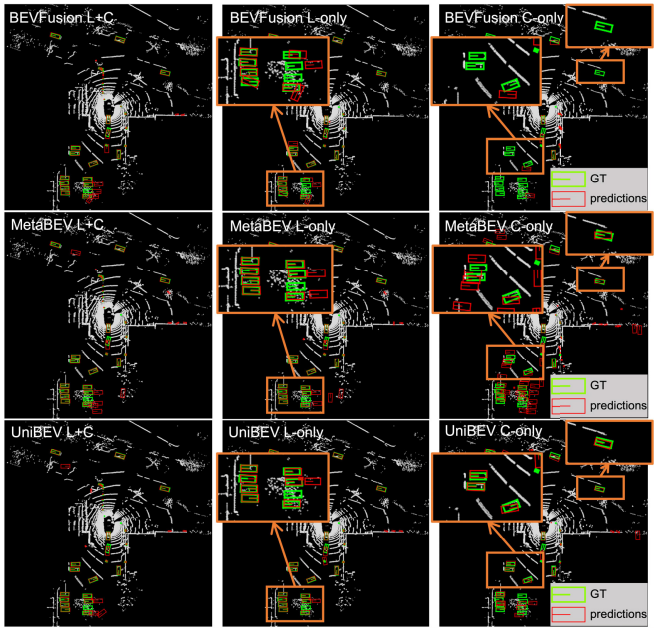

Fig. 4. **Qualitative detection results on nuScenes val set.** Columns: L+C, L, and C input. Rows: BEVFusion, MetaBEV, UniBEV (ours). Green boxes: ground truth; Red boxes: predictions. The key different zones are highlighted and zoomed in by orange boxes.

Fig. 5 (a) visualizes UniBEV's $N = 256$ normalized CNW weights, sorted from most LiDAR weighted to most camera weighted, and also reports their total summed weights. We observe the sum of camera weights is smaller than the sum of LiDAR weights ($106.1 < 149.9$). In other words, the learned fusion weights represent overall more reliance on LiDAR than on camera, which aligns with the overall better performance of LiDAR-only models over camera-only models. Still, we do observe quite diverse weight values. Certainly, not all channels favor LiDAR, and few weights are close to 0.5, the default for regular average fusion. The general higher influence of LiDAR on the fused results may also explain why the camera-only inference of UniBEV performs marginally worse than UniBEV_C, while the LiDAR-only inference of UniBEV even slightly outperforms UniBEV_L.

To validate CNW does not just scale channels to compensate for different magnitudes between LiDAR and camera BEV features, Fig. 5 (b) illustrates that the distribution of the average channel activations across the spatial map is the same for both modalities.

Inference speed. Finally, we measure the inference speed of all multi-modal methods using both input modalities. Using a Nvidia V100 GPU with a batch size of 1, we find that the average inference speed is 0.7 FPS for BEVFusion, 1.4 FPS for MetaBEV, and 1.6 FPS for our UniBEV. Thus, UniBEV achieves the highest speed of all multi-modal methods by a small margin, indicating its improved performance does not come at the cost of efficiency. We do note that BEVFusion uses a more powerful but slower backbone [3], [37]. Also, UniBEV's inference speed increases when running on multi-modal input, achieving 2.5 FPS for camera-only, and 3.9 FPS for LiDAR-only.

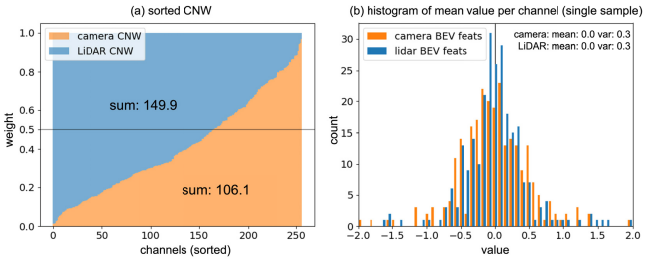

Fig. 5. (a) **Visualization of CNW's learned weights, with channels sorted by weight.** (b) **Histogram for the mean value per channel of camera and LiDAR BEV features from a single sample.** The histogram demonstrates the channel-wise alignment of the two BEV feature maps. Both modalities exhibit a mean and variance of 0 and 0.3, respectively.

TABLE II

COMPARISON OF DIFFERENT FUSION APPROACHES ON NUSCENES VAL SET FOR A FIXED DECODER DIMENSION OF 256. THE MODALITY DROPOUT STRATEGY IS APPLIED TO ALL MODELS. (**BEST**/ SECOND BEST)

Method	Encoder Dimensions	mAP L+C ↑	mAP L ↑	mAP C ↑	Summary Metric ↑
UniBEV_cat	$N/2 = 128$	63.8	57.6	34.4	51.9
UniBEV_avg	$N = 256$	64.1	57.6	**35.1**	52.3
UniBEV_CNW	$N = 256$	**64.2**	**58.2**	35.0	**52.5**

C. Ablation Study

We here discuss our ablation results for the different fusion modules, the effect of probabilities p_L and p_C during Modality Dropout, and the unified BEV queries.

*1) **Comparison of different fusion modules**:* We first test the performance of UniBEV with the different fusion strategies of Section III-C: concatenation (UniBEV_cat), average (UniBEV_avg) and CNW (UniBEV_CNW). Table II demonstrates that concatenation exhibits the lowest performance with a summary mAP of 51.9%. Since a missing modality results in concatenation filling multiple fused channels with zeros, such missing information cannot be compensated by the remaining sensor. Both UniBEV_avg and UniBEV_CNW avoid zero-filling in modality dropout, and subsequently elevate their performance to closely matched levels, achieving 52.3% and 52.5% summary mAP respectively.

When evaluating the performance across diverse input modalities, the L+C and L-only performances of UniBEV_CNW improve relative to UniBEV_avg, particularly evident in the L-only performance, but the C-only performance sees a decline. We hypothesize that CNW effectively lets the detector head rely more on LiDAR for the final fusion result, impacting its camera-only performance. Overall, the performance gap between CNW and average fusion appears minor, and if such a trade-off is favorable for the target application remains a future research.

*2) **Effect of probabilities p_L and p_C during Modality Dropout**:* Next, we investigate the influence of the probabilities of keeping the different modalities during MD. Table III showcases the performance of the model when changing p_L and p_C while keeping the probability of dropping a modality fixed to $p_{md} = 0.5$.

The L+C performance declines significantly in the two extreme cases where $p_L = 0$ and $p_L = 1$. However, in the other cases the L+C performance remains mostly stable, even when LiDAR-only and camera-only probabilities are unbalanced. As expected, the performance of LiDAR-only and camera-only consistently improves as the proportion of their respective inputs used during training increases.

As shown in the table, LiDAR-only mAP increases by 12.8 percentage points as the probability of LiDAR-only training rises from 0% to 75%, but already achieves an mAP of 45.5% even without LiDAR-only inputs during training. Remarkably, training with 100% LiDAR-only during MD decreases both LiDAR-only and L+C performance compared to including 25% camera-only training iterations, indicating that the camera-only input also regularizes the network for LiDAR.

The camera-only performance sees a substantial increase in mAP by 33 percentage points as the proportion of camera-only training iterations increases from 0% to 100%. However, unlike LiDAR, the camera-only performance can only reach 3.0% mAP without any training with camera-only inputs, and does not benefit from adding LiDAR-only training iterations.

These observations further support our insights, namely that fusion mostly relies on the more informative sensor, in this case LiDAR, which allows to train the LiDAR features through fusion even with few LiDAR-only training iterations. The same is not true for the camera features, which as the weaker modality strictly relies on MD to make the network achieve good performance. This observation also suggests that especially emphasizing the weaker modality during training could enhance the overall robustness of the model, even for the other modality. Due to the optimal summary and L+C performance, we keep $p_{md} = 0.5$ and $p_L = p_C = 0.5$ as the default for all our other experiments. We leave studying the impact of varying the MD probability p_{md} as future work.

TABLE III

EFFECT OF SENSOR DROPPING PROBABILITIES ON NUSCENES VAL SET. THE MD PROBABILITY p_{md} IS 0.5. (**BEST**/SECOND BEST/***default setup***)

		mAP			
p_L	p_C	L+C ↑	L ↑	C ↑	Summary Metric ↑
0	1	63.2	45.5	**36.0**	48.2
0.25	0.75	64.0	57.8	35.8	**52.5**
0.50	***0.50***	**64.2**	58.2	35.0	**52.5**
0.75	0.25	63.8	**58.3**	33.2	51.8
1	0	60.8	55.9	3.0	39.9

*3) **Unified Queries vs. Separate Queries**:* Finally, we compare the performance of UniBEV using shared BEV queries Q across modalities against a variant that learns separate queries for each modality. Table IV shows the model with unified queries has a minor edge over its counterpart with separate queries across all three input combinations, as well as in the summary metric (52.5% vs. 52.2% summary mAP). Interestingly, the model with separate queries matches the performance with UniBEV_L (refer to Table I, 57.8% mAP) while the unified queries model can slightly exceed this LiDAR-only trained counterpart. A possible explanation is that shared queries provide a weak interaction between the BEV encoders during training, which facilitates aligning their feature spaces. We conclude that the overall performance difference is only small, though the unified query design has the additional advantage that it requires fewer model parameters. For these reasons, we have adopted it as our default configuration.

TABLE IV

COMPARISON BETWEEN SEPARATE QUERIES DESIGN AND UNIFIED QUERIES DESIGN ON NUSCENES VAL SET. (**BEST**/SECOND BEST)

	mAP			
Method	L+C ↑	L ↑	C ↑	Summary Metric ↑
separate queries	64.0	57.8	34.9	52.2
unified queries	**64.2**	**58.2**	**35.0**	**52.5**

V. CONCLUSIONS

We have presented UniBEV, a multi-modal 3D object detection model designed with missing sensor modality inputs in mind. The experiments demonstrate UniBEV's higher robustness to missing inputs compared to SotA BEV-based detection methods, BEVFusion, and MetaBEV. UniBEV achieves 52.5% mAP on average over all input combinations, significantly improving over the baselines, with BEVFusion averaging at 43.5% mAP, and MetaBEV averaging at 48.7% mAP. Our proposed CNW fusion approach demonstrates superior performance compared to the commonly employed concatenation. The analysis of the learned weights reveals the inherent characteristics of a fusion process, notably that the model exhibits a greater reliance on LiDAR features as compared to camera features.

Future research can address dynamically adjusting channel weights, for instance based on the environmental conditions or content of the scene. Another open question remains what properties multi-modal features should exactly possess for robustness. Finally, we will explore if UniBEV's fused BEV features also benefit other tasks, such as BEV map segmentation. We hope that our work will inspire further research on robust perception for autonomous driving.

ACKNOWLEDGMENT

This work was supported by the 3D Urban Understanding (3DUU) Lab funded by the TU Delft AI Initiative.

REFERENCES

[1] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *CVPR*, 2020. 1

[2] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A multimodal dataset for autonomous driving," in *CVPR*, 2020. 1, 4

[3] T. Liang, H. Xie, K. Yu, Z. Xia, Z. Lin, Y. Wang, T. Tang, B. Wang, and Z. Tang, "BEVFusion: A simple and robust lidar-camera fusion framework," in *NeurIPS*, 2022. 1, 2, 3, 5, 6

[4] C. Ge, J. Chen, E. Xie, Z. Wang, L. Hong, H. Lu, Z. Li, and P. Luo, "MetaBEV: Solving sensor failures for bev detection and map segmentation," *arXiv preprint arXiv:2304.09801*, 2023. 1, 2, 4, 5

[5] Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. Rus, and S. Han, "BEVFusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," in *ICRA*, 2023. 1, 2, 3, 4

[6] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," in *ECCV*, 2020. 1, 2, 3, 4, 5

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017. 2

[8] Z. Chen, Z. Li, S. Zhang, L. Fang, Q. Jiang, F. Zhao, B. Zhou, and H. Zhao, "AutoAlign: Pixel-instance feature aggregation for multi-modal 3d object detection," in *IJCAI*, 2022. 2

[9] Z. Chen, Z. Li, S. Zhang, L. Fang, Q. Jiang, and F. Zhao, "AutoAlignV2: Deformable feature aggregation for dynamic multi-modal 3d object detection," in *ECCV*, 2022. 2

[10] X. Bai, Z. Hu, X. Zhu, Q. Huang, Y. Chen, H. Fu, and C.-L. Tai, "Transfusion: Robust lidar-camera fusion for 3d object detection with transformers," in *CVPR*, 2022. 2

[11] Y. Li, Z. Ge, G. Yu, J. Yang, Z. Wang, Y. Shi, J. Sun, and Z. Li, "BEVDepth: Acquisition of reliable depth for multi-view 3d object detection," *arXiv preprint arXiv:2206.10092*, 2022. 2

[12] Y. Li, A. W. Yu, T. Meng, B. Caine, J. Ngiam, D. Peng, J. Shen, Y. Lu, D. Zhou, Q. V. Le, *et al.*, "Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection," in *CVPR*, 2022. 2

[13] X. Chen, T. Zhang, Y. Wang, Y. Wang, and H. Zhao, "Futr3d: A unified sensor fusion framework for 3d detection," in *CVPR*, 2022. 2, 3, 4

[14] Y. Wang, V. C. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. Solomon, "Detr3d: 3d object detection from multi-view images via 3d-to-2d queries," in *CoRL*, 2022. 2, 4

[15] Y. Wang and J. M. Solomon, "Object dgcnn: 3d object detection using dynamic graphs," in *NeurIPS*, 2021. 2

[16] Z. Yang, J. Chen, Z. Miao, W. Li, X. Zhu, and L. Zhang, "Deepinteraction: 3d object detection via modality interaction," in *NeurIPS*, 2022. 2

[17] H. Hu, F. Wang, J. Su, Y. Wang, L. Hu, W. Fang, J. Xu, and Z. Zhang, "Ea-lss: Edge-aware lift-splat-shot framework for 3d bev object detection," *arXiv preprint arXiv:2303.17895*, 2023. 2

[18] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai, "BEVFormer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *ECCV*, 2022. 2, 3, 4, 5

[19] S. Borse, M. Klingner, V. R. Kumar, H. Cai, A. Almuzairee, S. Yogamani, and F. Porikli, "X-Align: Cross-modal cross-view alignment for bird's-eye-view segmentation," in *WCACV*, 2023. 2

[20] A. Saha, O. Mendez, C. Russell, and R. Bowden, "Translating images into maps," in *ICRA*, 2022. 2

[21] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *CVPR*, 2019. 2, 4, 5

[22] T. Yin, X. Zhou, and P. Krahenbuhl, "Center-based 3d object detection and tracking," in *CVPR*, 2021. 2, 4, 5

[23] F. Drews, D. Feng, F. Faion, L. Rosenbaum, M. Ulrich, and C. Gläser, "Deepfusion: A robust and modular 3d object detector for lidars, cameras and radars," in *IROS*, 2022. 2, 4

[24] B. Yang, M. Liang, and R. Urtasun, "Hdnet: Exploiting hd maps for 3d object detection," in *Conference on Robot Learning*, 2018. 2, 4

[25] J. Yan, Y. Liu, J. Sun, F. Jia, S. Li, T. Wang, and X. Zhang, "Cross modal transformer via coordinates encoding for 3d object dectection," *arXiv preprint arXiv:2301.01283*, 2023. 2, 3, 4

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016. 2, 4

[27] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3d object detection," in *CVPR*, 2018. 2, 4

[28] H. Cai, Z. Zhang, Z. Zhou, Z. Li, W. Ding, and J. Zhao, "BEVFusion4D: Learning lidar-camera fusion under bird's-eye-view via cross-modality guidance and temporal aggregation," *arXiv preprint arXiv:2303.17099*, 2023. 3

[29] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," in *ICLR*, 2020. 3

[30] Z. Xia, X. Pan, S. Song, L. E. Li, and G. Huang, "Vision transformer with deformable attention," in *CVPR*, 2022. 3

[31] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *ECCV*, 2020. 4

[32] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *CVPR*, 2017. 4

[33] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *ICCV*, 2021. 4

[34] B. Zhu, Z. Jiang, X. Zhou, Z. Li, and G. Yu, "Class-balanced grouping and sampling for point cloud 3d object detection," *arXiv preprint arXiv:1908.09492*, 2019. 5

[35] T. Wang, X. Zhu, J. Pang, and D. Lin, "Fcos3d: Fully convolutional one-stage monocular 3d object detection," in *ICCV*, 2021. 5

[36] M. Contributors, "MMDetection3D: OpenMMLab next-generation platform for general 3D object detection," https://github.com/open-mmlab/mmdetection3d, 2020. 5

[37] T. Liang, X. Chu, Y. Liu, Y. Wang, Z. Tang, W. Chu, J. Chen, and H. Ling, "Cbnet: A composite backbone network architecture for object detection," *IEEE T-IP*, 2022. 5, 6