Appendix

A Implementation Details

This section introduces the details utilized to implement LMD. First, we describe the dataset used for the BEV perception model employed to demonstrate our method, as well as the configuration of the fusion model utilized in the experiments.

A.1 Dataset

The nuScenes dataset [32] is a large-scale, multimodal dataset designed for autonomous driving tasks. It encompasses 1000 scenes, segmented into 700 for training, 150 for validation, and 150 for testing, each lasting about 20 seconds. The dataset captures a 360° horizontal field of view (FOV) through the use of six surround-view cameras, one lidar, and five radars, providing comprehensive environmental perception.

Notably, the nuScenes dataset stands out due to itshigh annotation frequency of every 0.5 seconds, translating to a 2 Hz rate, which includes over 1.4 million 3D bounding boxes across ten object categories. This extensive annotation facilitates advanced tasks like 3D object detection and tracking.

For evaluating detection performance, the dataset introduces unique metrics, including mean Average Precision (mAP) and five true positive (TP) metrics (ATE, ASE, AOE, AVE, and AAE) to measure errors in translation, scale, orientation, velocity, and attribute, respectively. These metrics are aggregated into the nuScenes Detection Score (NDS) to provide a comprehensive performance overview.

Furthermore, for the task like the Bird's Eye View (BEV) segmentation, the dataset follows specific settings proposed in prior research, leveraging its rich radar point cloud data among other modalities. The nuScenes dataset not only facilitates the development of advanced autonomous driving technologies but also sets a benchmark for evaluating the performance of these systems through its detailed and nuanced evaluation metrics.

A.2 Model Architecture & Hyper-Parameters

The basic settings of the camera-radar fusion model follow those of SimpleBEV [22]. The camera branch processes input from six view images, each of which is passed through a backbone network, such as ResNet [36], to extract image features. These image features, along with radar point cloud information, are subsequently mapped onto the BEV coordinate system to facilitate perception tasks in the BEV space. The camera features and radar features in the BEV representation are concatenated and processed using convolutional operations to generate latent features. In this paper, this convolution process is defined as the fusion operation performed by the layer function ($l = 1$). The latent features are then passed through a U-Net [37] decoder, after which the prediction head carries out a segmentation task focused on identifying vehicle regions. table 3 presents radar-camera and LiDAR-camera configuration details used for the SimpleBEV framework.

Table 3: Detailed configuration of modality-specific fusion setups

Parameter	**Radar - Camera**	**LiDAR - Camera**
Backbone	ResNet101	ResNet101
Fusion Operation	Concat & Conv	Concat & Conv
Sweeps	5	5
Use Meta Data	True	False
Input Size	(224, 400)	(224, 400)
BEV Coordinate	(200, 8, 200)	(200, 8, 200)

A.3 Experiments Compute Resources

We employed pre-trained encoders for camera, radar, and LiDAR modalities. Our method was evaluated for 6019 iterations with a batch size of 1 on 4 x NVIDIA RTX 3090 GPUs, which required approximately 3 hours in total.

B Proofs on LMD

B.1 Proposition 1: Handling Activation Layers

Let $A_j = F_j^{l-1}(x_c, x_r)$ denote the total input to the activation function for neuron j in layer l. Let $O_j = F_j^l(x_c, x_r)$ denote the original output of this activation function for neuron j in layer l. The linearized activation function for a modality-specific component h_{mj}^{l-1} is defined as $\hat{f}_j^l(h_{mj}^{l-1}) = c_j \cdot h_{mj}^{l-1}$. The constant c_j is given by $c_j = \frac{O_j}{A_j+\epsilon}$.

The total input to the activation function is the sum of its decomposed modality-specific components: $A_j = \sum_{m\in\{c,r,b\}} h_{mj}^{l-1}$. The output of the linearized layer $\hat{F}_j^l(x_c, x_r)$ is obtained by summing the outputs of the linearized activation function applied to each component (due to the linearity of $h \mapsto c_j h$ and as implied by Eq. (10) and (11) which state $h_{mj}^l = \hat{f}_j^l(h_{mj}^{l-1})$ and $\hat{f}_j^l(\sum_{m\in\{c,r,b\}} h_{mj}^{l-1}) = \sum_{m\in\{c,r,b\}} \hat{f}_j^l(h_{mj}^{l-1})$):

$$\hat{F}_j^l(x_c, x_r) = \sum_{m\in\{c,r,b\}} \hat{f}_j^l(h_{mj}^{l-1}) = \sum_{m\in\{c,r,b\}} c_j h_{mj}^{l-1} = c_j \sum_{m\in\{c,r,b\}} h_{mj}^{l-1} = c_j A_j.$$

Substituting the definition of c_j:

$$\hat{F}_j^l(x_c, x_r) = \left(\frac{O_j}{A_j+\epsilon}\right) A_j.$$

Eq (7) requires that the linearized model's output equals the original model's output at the operating point: $\hat{F}_j^l(x_c, x_r) = F_j^l(x_c, x_r)$, which is $\hat{F}_j^l(x_c, x_r) = O_j$. Therefore, we must have:

$$O_j = \left(\frac{O_j}{A_j+\epsilon}\right) A_j.$$

We analyze this equation in different cases:

- **Case 1:** $O_j \neq 0$**.** In this case, we can divide both sides by O_j:

  $$1 = \frac{A_j}{A_j+\epsilon}.$$

  This implies $A_j + \epsilon = A_j$, which means $\epsilon = 0$. Thus, if $O_j \neq 0$, eq. (7) is satisfied when $\epsilon = 0$.
- **Case 2:** $O_j = 0$**.** The equation becomes:

  $$0 = \left(\frac{0}{A_j+\epsilon}\right) A_j.$$

  This simplifies to $0 = 0$, provided that $A_j + \epsilon \neq 0$. This condition typically holds as ϵ is a small constant used to prevent division by zero. This case covers scenarios such as:
  - $A_j = 0$ and $O_j = 0$ (e.g., ReLU(0)=0, GELU(0)=0). Here $c_j = 0/(0+\epsilon) = 0$. Then $\hat{F}_j^l(x_c, x_r) = 0 \cdot 0 = 0$, which equals O_j. eq. (7) holds.
  - $A_j \neq 0$ but $O_j = 0$ (e.g., ReLU applied to a negative input). Here $c_j = 0/(A_j+\epsilon) = 0$ (assuming $A_j + \epsilon \neq 0$). Then $\hat{F}_j^l(x_c, x_r) = 0 \cdot A_j = 0$, which equals O_j. eq. (7) holds.

The connection to DTD, where c_j is interpreted as the slope of the segment joining (A_j, O_j) (and possibly the origin, if the activation passes through it), underpins this choice. If $c_j = O_j/A_j$ (for $A_j \neq 0$), then $\hat{F}_j^l = (O_j/A_j)A_j = O_j$, directly satisfying eq. (7). The formulation with ϵ makes this robust.

Thus, the proposed construction for the linearized activation function $\hat{f}_j^l$ ensures that eq. (7) is satisfied under the conditions discussed. □

B.2 Proposition 2: Handling BatchNorm Layers

The linearized output of the layer is the sum over modality components:

$$\hat{F}_j^l(x_c, x_r) = \sum_{m\in\{c,r,b\}} \hat{f}_j^l\big(h_{mj}^{l-1}\big).$$

Camera ($m = c$) and radar ($m = r$) components. Because $\delta_c = \delta_r = 0$,

$$\hat{f}_j^l\big(h_{mj}^{l-1}\big) = \frac{\gamma_j^l}{\sqrt{(\sigma_j^l)^2+\varepsilon}} h_{mj}^{l-1} \qquad (m \in \{c, r\}).$$

Bias component ($m = b$). With $\delta_b = 1$,

$$\hat{f}_j^l(h_{bj}^{l-1}) = \frac{\gamma_j^l}{\sqrt{(\sigma_j^l)^2 + \varepsilon}}\big(h_{bj}^{l-1} - \mu_j^l\big) + \beta_j^l.$$

Summing the three parts yields

$$\hat{F}_j^l(x_c, x_r) = \frac{\gamma_j^l}{\sqrt{(\sigma_j^l)^2 + \varepsilon}}\Big(h_{cj}^{l-1} + h_{rj}^{l-1} + h_{bj}^{l-1} - \mu_j^l\Big) + \beta_j^l.$$

Because the LMD framework guarantees $F_j^{l-1}(x_c, x_r) = \sum_{m\in\{c,r,b\}} h_{mj}^{l-1}$, we can rewrite the above as

$$\hat{F}_j^l(x_c, x_r) = \frac{\gamma_j^l}{\sqrt{(\sigma_j^l)^2 + \varepsilon}}\big(F_j^{l-1}(x_c, x_r) - \mu_j^l\big) + \beta_j^l,$$

which is the BatchNorm forward pass $F_j^l(x_c, x_r)$ in evaluation mode. Hence $\hat{F}_j^l = F_j^l$, satisfying eq. (7).

B.3 Proposition 3: Handling InstanceNorm Layers

Let $F^{l-1}(x_c, x_r)$ denote the input of the feature map to the InstanceNorm layer in layer l. For a specific channel j within this feature map, let H_j^{l-1} represent the activations for the channel j. The standard InstanceNorm operation for this channel is defined as:

$$F_j^l = \gamma_j \frac{H_j^{l-1} - \mathbb{E}[H_j^{l-1}]}{\sqrt{\text{Var}[H_j^{l-1}] + \epsilon}} + \beta_j$$

where $\mathbb{E}[H_j^{l-1}]$ and $\text{Var}[H_j^{l-1}]$ are the mean and variance of H_j^{l-1} computed over its spatial dimensions, respectively. γ_j and β_j are learnable scaling and shifting parameters for channel j.

In the LMD framework, the input to the l-th layer, $F^{l-1}(x_c, x_r)$, is decomposed into modality-specific components:

$$F^{l-1}(x_c, x_r) = \sum_{m\in\{c,r,b\}} h_m^{l-1}$$

where h_m^{l-1} represents the feature map attributed to the modality m (camera, radar, or bias) at layer $l-1$. For channel j, this means:

$$H_j^{l-1} = \sum_{m\in\{c,r,b\}} h_{mj}^{l-1}$$

The LMD framework requires that the linearized layer function $\hat{f}^l$ satisfies the property that the sum of the decomposed outputs equals the output of the linearized function applied to the sum of inputs, which in turn must match the original function output at the operating point from eq. (7).

For InstanceNorm, we state that the variance $\sigma_j^l = \text{Var}[H_j^{l-1}]$ is treated as a constant, saved from a single forward pass of the original fusion model.

By considering the proposed decomposition rule for $\hat{f}_j^l(h_{mj}^{l-1})$:

$$\hat{f}_j^l(h_{mj}^{l-1}) = \frac{h_{mj}^{l-1} - \mathbb{E}[h_{mj}^{l-1}]}{\sqrt{\sigma_j^l + \epsilon}}\gamma_j^l + \delta_m\beta_j^l$$

To sum these decomposed outputs over all modalities $m \in \{c, r, b\}$ to see if they reconstruct the original InstanceNorm output F_j^l:

$$\sum_{m\in\{c,r,b\}} \hat{f}_j^l(h_{mj}^{l-1}) = \sum_{m\in\{c,r,b\}} \left(\frac{h_{mj}^{l-1} - \mathbb{E}[h_{mj}^{l-1}]}{\sqrt{\sigma_j^l + \epsilon}}\gamma_j^l + \delta_m\beta_j^l\right)$$

By splitting the summation:

$$= \frac{\gamma_j^l}{\sqrt{\sigma_j^l + \epsilon}} \sum_{m \in \{c,r,b\}} (h_{mj}^{l-1} - \mathbb{E}[h_{mj}^{l-1}]) + \sum_{m \in \{c,r,b\}} \delta_m \beta_j^l$$

For the first term, we use the linearity of expectation:

$$\sum_{m \in \{c,r,b\}} (h_{mj}^{l-1} - \mathbb{E}[h_{mj}^{l-1}]) = \sum_{m \in \{c,r,b\}} h_{mj}^{l-1} - \sum_{m \in \{c,r,b\}} \mathbb{E}[h_{mj}^{l-1}]$$

$$= H_j^{l-1} - \mathbb{E}\left[\sum_{m \in \{c,r,b\}} h_{mj}^{l-1}\right]$$

$$= H_j^{l-1} - \mathbb{E}[H_j^{l-1}]$$

For the second term, within the definition of δ_m: $\delta_m = 0$ for $m \in \{c, r\}$ and $\delta_m = 1$ for $m \in \{b\}$. Therefore:

$$\sum_{m \in \{c,r,b\}} \delta_m \beta_j^l = \delta_c \beta_j^l + \delta_r \beta_j^l + \delta_b \beta_j^l = 0 \cdot \beta_j^l + 0 \cdot \beta_j^l + 1 \cdot \beta_j^l = \beta_j^l$$

Substituting these back into the summed expression:

$$\sum_{m \in \{c,r,b\}} \hat{f}_j^l(h_{mj}^{l-1}) = \frac{\gamma_j^l (H_j^{l-1} - \mathbb{E}[H_j^{l-1}])}{\sqrt{\sigma_j^l + \epsilon}} + \beta_j^l$$

Since σ_j^l is taken as the fixed variance $\mathrm{Var}[H_j^{l-1}]$ from the first forward pass, this expression is the original InstanceNorm output F_j^l. Since each modality is processed independently except for the shared, fixed scale, the resulting features adhere to the intended *separation property*. The proposed decomposition rule for InstanceNorm layers ensures that the sum of the modality-specific outputs from the linearized layer perfectly reconstructs the output of the original InstanceNorm layer, satisfying eq. (7) of the LMD framework.

C Variants on Activation Layers

Neccessities of considering the bias-splitting rules in activation layers, aside from cases where a constant term is inevitably introduced (e.g., normalization layers), is as follows. When a model is linearized, a bias' prediction can be obtained by inputting zeros into all data points. As shown in fig. 2, it is evident that bias features also exhibit perception capabilities. Therefore, the process of appropriately distributing the bias contribution between the camera and radar can be considered. Particularly in the activation layers, if a specific neuron is activated by the camera, adding a positive bias to the camera feature at that neuron could enhance the camera's contribution. Conversely, if a neuron is activated by the radar, adding a positive bias to the radar feature could strengthen the radar's contribution. This idea led to the development of the sum rule introduced in appendix C.1. The modality predictions based on various bias-splitting rules need to be extensively explored in future experiments.

While the sum rule adds the entire bias to whichever modality actually triggers a ReLU, it disregards how strongly the two modalities contribute in magnitude. Empirically this may still blur modality separation when both camera and radar deliver noticeable—but differently scaled—signals. The ratio splitting rule therefore apportions the bias term h_{bj}^{l-1} proportionally to the absolute pre-activations of camera and radar. The impact of sum and ratio splitting rules can be assessed via the four perturbation-based metrics (R_p/R, R_p/C, C_p/R, C_p/C). These rules modulate how sensitively and independently each modality responds to perturbation, offering an empirical basis for bias handling in activation layers.

C.1 Sum Splitting Rule

For the most common activation function, ReLU, if a camera feature value is less than zero but a radar feature value is greater than zero, resulting in the activation of a specific neuron, it is reasonable to interpret that the neuron was activated by the radar. In such cases, a splitting rule can be applied where the bias feature is added to the radar feature when the bias feature value is greater than zero. Conversely, if both a camera feature value and a bias feature value are greater than zero while the radar feature value is less than zero, distributing the bias feature to the camera feature strengthen the camera's contribution to that neuron. The sum condition tested in our experiments are derived from this idea. The related equations are as follows:

$$
\begin{aligned}
Cam - Condition &= h_{cj}^{l-1} < 0, h_{rj}^{l-1} > 0, h_{bj}^{l-1} < 0 \\
&\quad \text{and } h_{cj}^{l-1} > 0, h_{rj}^{l-1} < 0, h_{bj}^{l-1} > 0 \\
Rad - Condition &= h_{cj}^{l-1} < 0, h_{rj}^{l-1} > 0, h_{bj}^{l-1} > 0 \\
&\quad \text{and } h_{cj}^{l-1} > 0, h_{rj}^{l-1} < 0, h_{bj}^{l-1} < 0 \\
\hat{f}^l(h_{cj}^{l-1}) &= \begin{cases} c\left(h_{cj}^{l-1} + h_{bj}^{l-1}\right), & \text{for } Cam - Condition, \\ c\left(h_{cj}^{l-1}\right), & \text{otherwise.} \end{cases} \\
\hat{f}^l(h_{rj}^{l-1}) &= \begin{cases} c\left(h_{rj}^{l-1} + h_{bj}^{l-1}\right), & \text{for } Rad - Condition, \\ c\left(h_{rj}^{l-1}\right), & \text{otherwise.} \end{cases} \\
\hat{f}^l(h_{bj}^{l-1}) &= \begin{cases} 0, & \text{for } Cam \,\&Rad - Condition, \\ c\left(h_{bj}^{l-1}\right), & \text{otherwise.} \end{cases} \\
\text{, where} \quad c &= \frac{F_j^l(\mathbf{x_c}, \mathbf{x_r})}{F_j^{l-1}(\mathbf{x_c}, \mathbf{x_r}) + \epsilon}.
\end{aligned}
\tag{13}
$$

C.2 Ratio Splitting Rule

In activation layers a bias term h_{bj}^{l-1} can enhance the modality responsible for the neuron's activation. The ratio splitting rule redistributes that bias to the camera and radar features in proportion to their absolute magnitudes, while leaving neurons that fail the ratio conditions unchanged. This preserves the conservation property in eq. (7) and generalizes the identity and uniform rules introduced earlier.

$$
\begin{aligned}
Ratio - Condition &= \big(h_{cj}^{l-1} > 0,\, h_{rj}^{l-1} > 0,\, h_{bj}^{l-1} > 0\big) \\
&\quad \text{or } \big(h_{cj}^{l-1} < 0,\, h_{rj}^{l-1} < 0,\, h_{bj}^{l-1} < 0\big), \\
Ratio - Condition2 &= \big(h_{cj}^{l-1} > 0,\, h_{rj}^{l-1} > 0,\, h_{bj}^{l-1} < 0\big) \\
&\quad \text{or } \big(h_{cj}^{l-1} < 0,\, h_{rj}^{l-1} < 0,\, h_{bj}^{l-1} > 0\big).
\end{aligned}
$$

$$
\alpha_j \;=\; \frac{|h_{rj}^{l-1}|}{|h_{cj}^{l-1}| + |h_{rj}^{l-1}| + \varepsilon}, \qquad \alpha_j \in [0,1],
$$

with a small $\varepsilon > 0$ for numerical stability.

Let $c = \dfrac{F_j^l(x_c, x_r)}{F_j^{l-1}(x_c, x_r) + \varepsilon}$ be the slope from proposition 1. Then

$$
\hat{f}^{\,l}(h_{cj}^{l-1}) = \begin{cases} c\big(h_{cj}^{l-1} + (1-\alpha_j)\, h_{bj}^{l-1}\big), & Ratio - Condition, \\ c\big(h_{cj}^{l-1} + \alpha_j\, h_{bj}^{l-1}\big), & Ratio - Condition2, \\ c\, h_{cj}^{l-1}, & \text{otherwise,} \end{cases}
$$

$$
\hat{f}^{\,l}(h_{rj}^{l-1}) = \begin{cases} c\big(h_{rj}^{l-1} + \alpha_j\, h_{bj}^{l-1}\big), & Ratio - Condition, \\ c\big(h_{rj}^{l-1} + (1-\alpha_j)\, h_{bj}^{l-1}\big), & Ratio - Condition2, \\ c\, h_{rj}^{l-1}, & \text{otherwise,} \end{cases}
$$

$$
\hat{f}^{\,l}(h_{bj}^{l-1}) = \begin{cases} 0, & Ratio - Conditionor Ratio - Condition2, \\ c\, h_{bj}^{l-1}, & \text{otherwise.} \end{cases}
$$

Because h_{bj}^{l-1} is only re-partitioned, the eq. (7) is strictly preserved.

The ratio rule thus interpolates smoothly between existing bias-handling strategies while adapting to the magnitude of each modality's evidence, leading to the stability improvements reported in section 5.

Table 4: Comparison of Pearson Correlation and Mean Squared Error for selected LMD rules (Radar + Camera)

Modality	Method	Pearson Correlation				Mean Squared Error			
		R_p/R (↓)	R_p/C (↑)	C_p/R (↑)	C_p/C (↓)	R_p/R (↑)	R_p/C (↓)	C_p/R (↓)	C_p/C (↑)
	Activation								
Radar + Camera	Sum-Splitting	0.12 ± 0.04	1.00 ± 0.00	1.00 ± 0.00	0.23 ± 0.05	**9.63** ± 2.34	0.00 ± 0.00	0.00 ± 0.00	**40.43** ± 19.35
	Ratio-Splitting	**0.08** ± 0.06	**1.00** ± 0.00	**1.00** ± 0.00	**0.18** ± 0.04	4.46 ± 1.45	**0.00** ± 0.00	**0.00** ± 0.00	30.35 ± 4.19

D Further Experiments

D.1 Further Post-hoc Interpretation

This section presents additional examples of applying LMD to the BEV perception task. Through further visualizations, we expect that our proposed method is validated with sufficient consistency to reliably interpret the model.

fig. 3 and fig. 4 show further visualization results for the reliability of the proposed LMD on the BEV perception benchmark. fig. 3 normalizes each modality-specific feature map with a sigmoid transform before projection, so only positive activations remain; the bright regions therefore mark locations where a single sensor delivers decisive, supportive cues to the detector, making the saliency of true objects immediately apparent. By contrast, fig. 4 retains the full signed output and colour-codes positive and negative responses, allowing us to see not only where a modality reinforces the final prediction but also where it actively suppresses prediction.

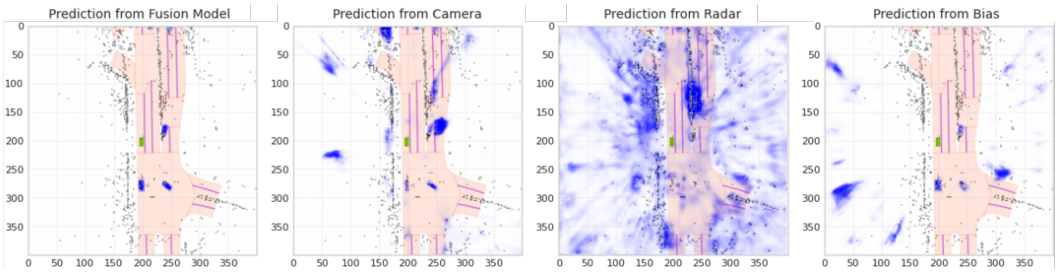

Figure 3: Visualizations of using Sigmoid.

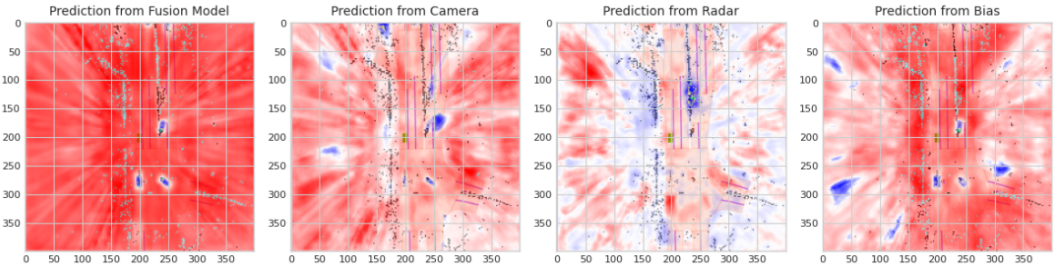

Figure 4: Visualizations of both Positive and Negative Values.

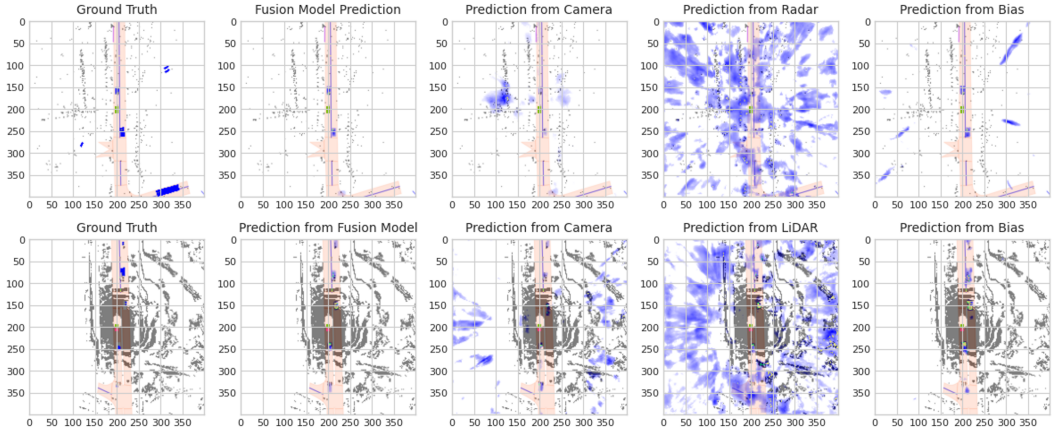

Figure 5: Further Qualitative Visualization Results

D.2 Metric Formulas

To evaluate the separation property of the modality-specific decomposition, we introduce four perturbation-based metrics. Here, perturbation refers to the replacement of one modality input with an uncorrelated sample. These metrics assess how much each modality-specific output responds to changes in its own input and remains invariant to others. Each metric is computed using the Pearson Correlation Coefficient (PCC) and Mean Squared Error (MSE).

Pearson Correlation Coefficient (PCC). Briefly, for pearson correlation coefficient, we define:

$$\mathrm{PCC}_{k,m} = \frac{\sum_i \left(P_k[i] - \bar{P}_k\right)\left(P_{(k+500m)\bmod N}[i] - \bar{P}_{(k+500m)\bmod N}\right)}{\sqrt{\sum_i \left(P_k[i] - \bar{P}_k\right)^2 \sum_i \left(P_{(k+500m)\bmod N}[i] - \bar{P}_{(k+500m)\bmod N}\right)^2}}, \tag{14}$$

where $N = 6019$, P_k and $P_{(k+500m)\bmod N}$ are the predictions for the sample spaced by $500m$ from k, $\bar{P}_k$ and $\bar{P}_{(k+500m)\bmod N}$ are their respective mean values, and $m \in \{1, 2, \ldots, 12\}$ ensures 12 distinct comparisons. PCC measures the similarity between two prediction vectors by first subtracting their respective means (mean-centering), computing the dot product of the centered vectors, and normalizing it by the product of their standard deviations. The resulting value lies in the range $[-1, 1]$, where 1 indicates perfect correlation, 0 indicates no correlation, and -1 indicates perfect inverse correlation.

Mean Squared Error (MSE). In addition to PCC, we compute the mean squared error to quantify absolute differences:

$$\mathrm{MSE}_{k,m} = \frac{1}{M}\sum_{i=1}^{M}(P_k[i] - P_{(k+500m)\bmod N}[i])^2, \tag{15}$$

where M denotes the total number of pixels in the BEV feature map.

Perturbation-based Metrics. Let x_c and x_r denote the clean camera and radar inputs, and x_c^{pert}, x_r^{pert} their respective perturbed versions. Let $F_{\text{camera}}(x_c, x_r)$ and $F_{\text{radar}}(x_c, x_r)$ denote the modality-specific predictions.

We define the following four metrics:

- R_p/R**.** Radar prediction change under radar perturbation:
  $$\mathrm{PCC}_{k,m}(F_{\text{radar}}(x_c, x_r^{\text{pert}}), F_{\text{radar}}(x_c, x_r)), \quad \mathrm{MSE}_{k,m}(F_{\text{radar}}(x_c, x_r^{\text{pert}}), F_{\text{radar}}(x_c, x_r))$$
  Lower PCC and higher MSE are better, indicating sensitivity to radar input.
- R_p/C**.** Camera prediction change under radar perturbation:
  $$\mathrm{PCC}_{k,m}(F_{\text{camera}}(x_c, x_r^{\text{pert}}), F_{\text{camera}}(x_c, x_r)), \quad \mathrm{MSE}_{k,m}(F_{\text{camera}}(x_c, x_r^{\text{pert}}), F_{\text{camera}}(x_c, x_r))$$
  Higher PCC and lower MSE are better, indicating invariance of the camera modality.
- C_p/R**.** Radar prediction change under camera perturbation:
  $$\mathrm{PCC}_{k,m}(F_{\text{radar}}(x_c^{\text{pert}}, x_r), F_{\text{radar}}(x_c, x_r)), \quad \mathrm{MSE}_{k,m}(F_{\text{radar}}(x_c^{\text{pert}}, x_r), F_{\text{radar}}(x_c, x_r))$$
  Higher PCC and lower MSE are better, indicating invariance of the radar modality.
- C_p/C**.** Camera prediction change under camera perturbation:
  $$\mathrm{PCC}_{k,m}(F_{\text{camera}}(x_c^{\text{pert}}, x_r), F_{\text{camera}}(x_c, x_r)), \quad \mathrm{MSE}_{k,m}(F_{\text{camera}}(x_c^{\text{pert}}, x_r), F_{\text{camera}}(x_c, x_r))$$
  Lower PCC and higher MSE are better, indicating sensitivity to camera input.

These metrics are averaged over the entire test set using uniformly sampled perturbations. Together, they verify whether the decomposition satisfies: (1) sensitivity to the perturbed modality and (2) invariance to the unperturbed modality — key indicators of successful modality separation.

E Discussion

E.1 Broader Impact

LMD introduces the first post-hoc framework capable of attributing the perdictions of each layer in a multi-sensor fusion model to its respective modalities, without modifying the original architecture.

Positive societal impacts may include:

- **Enhanced safety in autonomous systems.** By revealing the sensor modality (camera, LiDAR, or radar) primarily responsible for each prediction, LMD enables identification of single-sensor failures before they propagate into critical errors.

- **Accelerated certification and debugging.** Modality-level attributions can support system audits and facilitate the generation of safety evidence, in line with emerging regulatory requirements for ADAS systems.

- **Extension of interpretability to other multi-modal domains.** Since LMD is agnostic to model architecture, it can be applied to a wide range of multimodal perception systems beyond autonomous driving, such as medical image–text fusion or surveillance video analysis, to clearly attribute the role of each input modality in the model's decision-making process.

- **Commitment to open science.** We will publicly release our codebase, pretrained checkpoints, and evaluation scripts under an open-source license to promote reproducibility and rigorous external validation.

Potential risks include:

- **Misleading reassurance.** Users may over-trust visual explanations; a clear modality-wise attribution does not guarantee that the fused decision is accurate.

- **Bias amplification.** LMD uncovers but does not correct dataset imbalances, and naive use of its explanations may perpetuate structural biases.

- **Residual opacity.** The interpretability of high-order interactions and root-point selections in activation layers remains an open challenge for future work.

To mitigate these risks, we recommend pairing LMD with the perturbation-based metrics introduced in this paper to quantitatively assess explanation fidelity prior to deployment. We also encourage conducting independent reproducibility audits on geographically and demographically diverse datasets, and suggest applying the same level of scrutiny—augmented by domain-specific expertise and operational context—when extending LMD to other multimodal domains, such as medical imaging or service robotics. LMD sets a new benchmark for interpretability in multimodal domains and provides a solid foundation for future research and real-world deployment.

E.2 Model-Wise Explanation

Understanding the extent to which we can perform model-wise explanations by comparing the camera-only model and the fusion model is a critical issue. Interpreting the contributions of each modality based on the results of these two models for specific data is highly challenging. This difficulty arises, for the two models are trained on different input distributions.

For instance, if a fusion model successfully detects an object that camera-only model fails to detect, it would be a clear error to attribute this success to radar data. Predictions from the camera-only model should be utilized only for references. For example, if the camera-only model successfully detects a particular vehicle, it can be inferred that the information captured by the camera sensor for that region is sufficient. This insight can be considered when analyzing the fusion model using the LMD approach.

Nonetheless, it is crucial to recognize that LMD is designed to provide post-hoc interpretations for a single trained model, not to be a tool for interpreting two different models trained on different data distributions.

E.3 GradCAM Experiments

Since LMD enables access to internal activation maps, it is possible to compute activation maps from radar data and subsequently apply the Grad-CAM [38] approach.

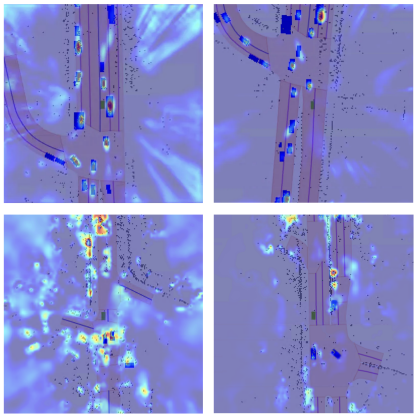

Figure 6: Visualizations of Radar-GradCAM Using Radar's Activations from LMD.

Grad-CAM [38] leverages gradient information to highlight relevant areas, Grad-CAM++ [39] introduces modified approach by taking positive gradients of target functions, and Score-CAM [40] bypasses the need for gradient information altogether, instead relying on activation scores to determine feature significance. Seg-Grad-CAM [41] is an adaptation of the Grad-CAM method, applied to the task of image segmentation by treating the segmentation output as the target function while maintaining the original mechanism of Grad-CAM for visual explanations. This approach enables a focused analysis of regions relevant to segmentation tasks, leveraging the gradients of the target output with respect to the feature maps. The Grad-CAM is calculated by :

$$L^c_{\text{Grad-CAM}} = \text{ReLU}\left(\sum_k \alpha^c_k A^{lk}\right) \tag{16}$$

where $L^c_{\text{Grad-CAM}}$ is the class activation map for class c, α^c_k are the weights for the k-th feature map in l-th layer A^{lk}, computed as the global average of the gradients flowing back from the output unit for class c. ReLU is applied to focus on features that have a positive influence on the class of interest. The weight of k-th feature map for class c is computed as follows.

$$\alpha^c_k = \frac{1}{Z}\sum_i \sum_j \frac{\partial y^c}{\partial A^{lk}_{ij}} \tag{17}$$

where y^c, A^{lk}_{ij}, and Z denote the score for class c, (i, j)-th entry in k-th activation map, and normalization factor respectively. In Seg-Grad-CAM, y^c is replaced by $\sum_{(i,j)\in R} y^c_{ij}$, where R is a set of pixel indices of interest in the output mask.

Through obtained activations which are derived solely from the radar data with LMD, it becomes possible to use the conventional CAM method based on the activations at the intermediate layer, which we call Radar-GradCAM. The Radar-GradCAM is calculated by :

$$L^c_{\text{Radar-GradCAM}} = \text{ReLU}\left(\sum_k \alpha^c_k (\hat{A}^{lk}(\mathbf{o_c}, \mathbf{x_r}) - \hat{A}^{lk}(\mathbf{o_c}, \mathbf{o_r}))\right) \tag{18}$$

, where $\hat{A}^{lk}(\mathbf{o_c}, \mathbf{x_r})$ and $\hat{A}^{lk}(\mathbf{o_c}, \mathbf{o_r})$ are the activations of linearized fusion model $\hat{F}$ and α^c_k is calculated in the same manner as in eq. (17).

$$\alpha^c_k = \frac{1}{Z}\sum_i \sum_j \frac{\partial \sum_{(i,j)\in R} \hat{F}^l_j(\mathbf{x_c}, \mathbf{x_r})}{\partial A^{lk}_{ij}(\mathbf{x_c}, \mathbf{x_r})} \tag{19}$$

Based on this calculation, the Radar-GradCAM shown in fig. 6 reveals which parts of the fusion model are relevant to the radar data points, by taking the radar-only activations that are crucial to the perception task.

F Extention to General Multimodal Model

In this Section, we demonstrate that the LMD method can be generalized to multimodal models. We achieve this by extending the formulation to accommodate M modalities

Beginning at the fusion layer ($l = 1$), the first-order Taylor expansion of f_j^1 becomes :

$$f_j^1(\mathbf{x_{m_1}}, \ldots, \mathbf{x_{m_M}}) = \sum_i \sum_{m \in \{m_1, \ldots, m_M\}} J_{mji}^1 x_{mi} + \underbrace{\left(f_j^1(\tilde{x}_{m_1}, \ldots, \tilde{x}_{m_M}) - \sum_i \sum_{m \in \{m_1, \ldots, m_M\}} J_{mji} \tilde{x}_{mi} + \epsilon \right)}_{b_j^1} \tag{20}$$

where m indexes the modality, $\tilde{x}_{m_1}, \ldots, \tilde{x}_{m_M}$ represent reference points for M modalities, and $\mathbf{J}_{mji}$ denotes the Jacobian calculated at the reference points for modality m.

For subsequent layer with $l = 2, \ldots, N$, suppose l-th layer function takes $\sum_{m \in \{m_1, \ldots, m_M, b\}} h_{mj}^{l-1}$ as inputs, the following holds :

$$f_j^l \left(\sum_{m \in \{m_1, \ldots, m_M, b\}} h_{mj}^{l-1} \right) = \sum_i \sum_{m \in \{m_1, \ldots, m_M, b\}} J_{mji}^{l-1}(h_{mi}^{l-1}) + b_j^l = \sum_{m \in \{m_1, \ldots, m_M, b\}} h_{mj}^l \tag{21}$$

We define all modality features that all layers in fusion model be decomposed into them.

$$F_j^l(\mathbf{x_{m_1}}, \ldots, \mathbf{x_{m_M}}) = f_j^l \left(\sum_{m \in \{m_1, \ldots, m_M, b\}} h_{mj}^{l-1} \right) = h_{m_1 j}^l + \cdots + h_{m_M j}^l + h_{bj}^l \tag{22}$$

From eq. (22), we can confirm inductively that the following holds. For the linearized layer functions $\hat{f}^1, \ldots, \hat{f}^N$ with bias-splitting rules applied, $F_j^l(\mathbf{x_{m_1}}, \ldots, \mathbf{x_{m_M}}) = \hat{F}_j^l(\mathbf{x_{m_1}}, \ldots, \mathbf{x_{m_M}})$ for $l \in \{1, \ldots, N\}$, where $\hat{F}^l(\mathbf{x_{m_1}}, \ldots, \mathbf{x_{m_M}})$ be $\hat{f}^l \circ \cdots \circ \hat{f}^2 \circ \hat{f}^1(\mathbf{x_{m_1}}, \ldots, \mathbf{x_{m_M}})$, forcing the epsilon in eq. (20) to zero. Based on all the layer functions, $\hat{f}^1, \ldots, \hat{f}^N$, the modality features are computed as follows :

$$h_{mj}^l = \hat{f}_j^l(h_{mj}^{l-1}) \,,\ m \in \{m_1, \ldots, m_M, b\} \tag{23}$$

Furthermore, the following arithmetic is satisfied for layer function $\hat{f}_j^l$:

$$\hat{f}_j^l \left(\sum_{m \in \{m_1, \ldots, m_M, b\}} h_{mj}^{l-1} \right) = \hat{f}_j^l(h_{m_1 j}^{l-1}) + \cdots + \hat{f}_j^l(h_{m_M j}^{l-1}) + \hat{f}_j^l(h_{bj}^{l-1}) \tag{24}$$

From eq. (23) and eq. (24), the following property holds for every layers :

$$\begin{aligned} \hat{F}_j^l(\mathbf{x_{m_1}}, \ldots, \mathbf{x_{m_M}}) = & \underbrace{\hat{F}_j^l(\mathbf{x_{m_1}}, \mathbf{o_{m_2}}, \ldots, \mathbf{o_{m_M}})}_{h_{m_1 j}^l} \\ & + \cdots + \underbrace{\hat{F}_j^l(\mathbf{o_{m_1}}, \ldots, \mathbf{o_{m_{M-1}}}, \mathbf{x_{m_M}})}_{h_{m_M j}^l} \\ & + \underbrace{\hat{F}_{m_M}^l(\mathbf{o_{m_1}}, \ldots, \mathbf{o_{m_M}})}_{h_{bj}^l} . \end{aligned} \tag{25}$$

The linearization and bias-splitting rules applied to the normalization and activation layers are consistent with those described in the main paper.