Uncertainty-Encoded Multi-Modal Fusion for Robust Object Detection in Autonomous Driving

Lou, Yang; Song, Qun; Xu, Qian; Tan, Rui; Wang, Jianping

DOI
10.3233/FAIA230441

Publication date
2023

Document Version
Final published version

Published in
ECAI 2023 - 26th European Conference on Artificial Intelligence, including 12th Conference on Prestigious Applications of Intelligent Systems, PAIS 2023 - Proceedings

ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230441

Uncertainty-Encoded Multi-Modal Fusion for Robust Object Detection in Autonomous Driving

Yang Lou[1], Qun Song[2], Qian Xu[1], Rui Tan[3] and Jianping Wang[1]

Abstract. Multi-modal fusion has shown initial promising results for object detection of autonomous driving perception. However, many existing fusion schemes do not consider the quality of each fusion input and may suffer from adverse conditions on one or more sensors. While *predictive uncertainty* has been applied to characterize single-modal object detection performance at run time, incorporating uncertainties into the multi-modal fusion still lacks effective solutions due primarily to the uncertainty's cross-modal incomparability and distinct sensitivities to various adverse conditions. To fill this gap, this paper proposes *Uncertainty-Encoded Mixture-of-Experts* (UMoE) that explicitly incorporates single-modal uncertainties into LiDAR-camera fusion. UMoE uses individual expert network to process each sensor's detection result together with encoded uncertainty. Then, the expert networks' outputs are analyzed by a gating network to determine the fusion weights. The proposed UMoE module can be integrated into any proposal fusion pipeline. Evaluation shows that UMoE achieves a maximum of 10.67%, 3.17%, and 5.40% performance gain compared with the state-of-the-art proposal-level multi-modal object detectors under extreme weather, adversarial, and blinding attack scenarios.

1 Introduction

Perception is a core subsystem of autonomous driving (AD) where onboard sensors such as LiDAR, camera, and radar are used to sense the surrounding environment. Object detection is one of the most critical perception tasks which localizes and identifies the objects of interest as important prerequisites to autonomous navigation. Recently, multi-modal fusion-based AD object detection has received enormous attention from both academia [6] and industry [31, 25]. In particular, as LiDAR and camera provide fundamentally different and complementary information about the objects (i.e., depth and visual features), the fusion based on these two modalities has shown initial promising results for object detection [12].

Recently, *predictive uncertainty* is proposed to measure the variability of model predictions under plausible inputs [15]. It has been used to measure the quality of single-modal object detection results [7, 22]. In the context of multi-modal fusion, we conjecture that the uncertainty regarding the sensing result in each modality is valuable to the fusion. For instance, when a sensor experiences transient interference, the resulting high uncertainty value is an important indicator for tuning down the weight of the corresponding sensing result in the fusion. However, incorporating uncertainty into fusion-based AD object detection has not received systematic study. Most existing LiDAR-camera fusion approaches [26, 34, 23] do not consider uncertainties. They may yield performance degradation when a sensor experiences sensing quality drop in certain adverse settings. The study in [5] is the only work considering uncertainty in fusion-based object detection. However, it only considers the uncertainty of LiDAR's sensing result and does not incorporate multi-modal uncertainty into the fusion-based object detection algorithm. Thus, it falls short of addressing the scenarios adverse on camera.

This paper aims at advancing the state of the art by designing LiDAR-camera fusion for AD object detection with each modality's predictive uncertainty incorporated. However, this turns out to be challenging, because i) there lacks informative and practical uncertainty representations, ii) the LiDAR's and camera's uncertainties, as dimensionless quantities, are not directly comparable, and iii) their sensitivities to various adverse conditions are greatly different. These properties render straightforward ways of incorporating uncertainties, e.g., admitting raw uncertainties as fusion inputs, futile.

To address the challenges, we propose a new multi-modal fusion module called *Uncertainty-Encoded Mixture-of-Experts* (UMoE) for robust AD object detection. First, UMoE applies the Monte Carlo Dropout [9] and Direct Modeling [15] approaches to estimate each sensor's uncertainty. Then, individual expert network is used to process each sensor's detection result with uncertainty encoded. Lastly, the output features of each expert network are analyzed by a gating network to determine the weights for fusion. With the uncertainty encoding for both modalities, UMoE can retain the object detection performance or allow more graceful performance degradation when a single sensor or both sensors suffer sensing quality drops in adverse scenarios.

This paper's contributions are summarized as follows:

- We identify the challenges caused by the cross-modality properties of uncertainty in the multi-modal fusion design, i.e., distinct uncertainty value ranges and varied sensitivities under different adverse conditions. Based on the understanding, we encode the LiDAR and camera uncertainties into comparable scores that can be leveraged to refine detections across modalities.
- We propose UMoE that applies encoded uncertainties to weigh and fuse the two sensing modalities for robust AD object detection. As a desirable feature, the UMoE module can be incorporated into any proposal-level fusion methods. To the best of our knowledge, UMoE is the first modularized mechanism incorporating multi-modal uncertainties into AD object detection.
- Experiments show that UMoE outperforms advanced and state-of-the-art LiDAR-camera fusion models on real-world and synthetic datasets, including clear, snowy, foggy, adversarial, and blinding attack scenarios.

[1] City University of Hong Kong, email: yanglou3-c@my.cityu.edu.hk, {qian.xu, jianwang}@cityu.edu.hk.
[2] Delft University of Technology, email: q.song-1@tudelft.nl.
[3] Nanyang Technological University, email: tanrui@ntu.edu.sg.

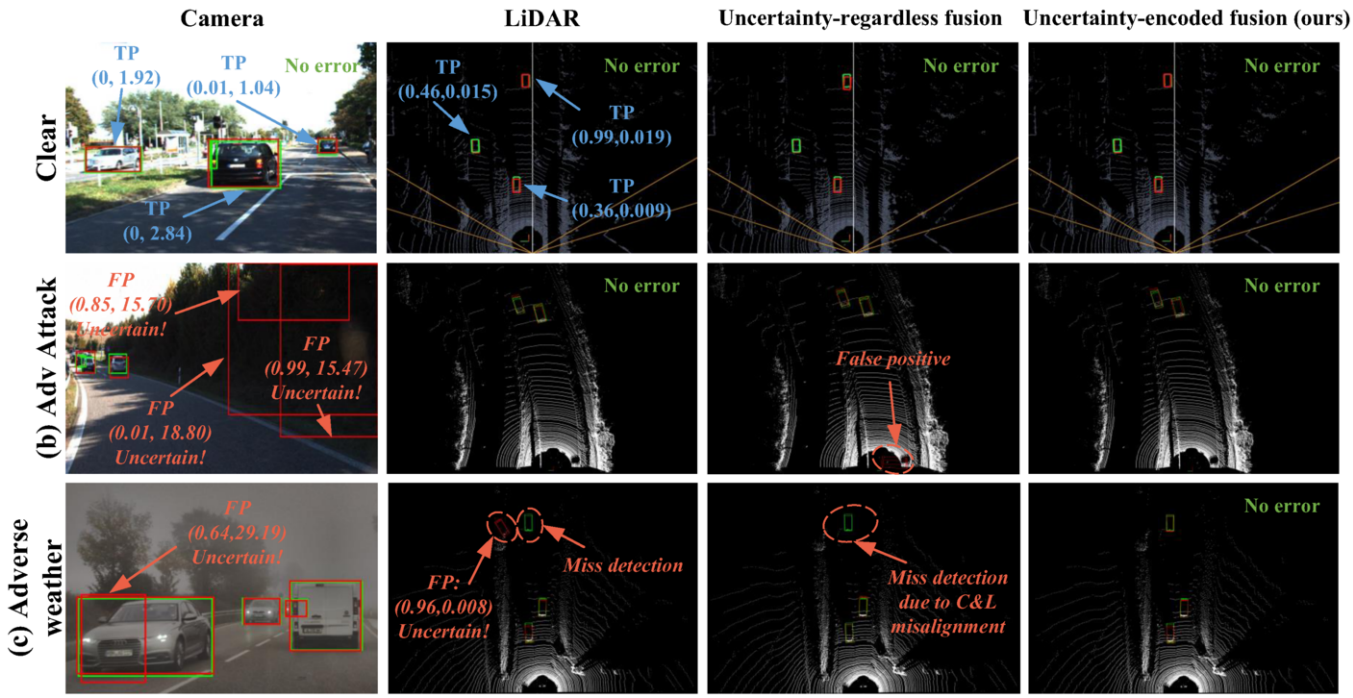


Figure 1. Object detection by camera, LiDAR, uncertainty-regardless fusion, and our uncertainty-encoded fusion under the three driving scenes of clear, adversarial attack on camera, and adverse weather condition. Green and red bounding boxes indicate ground-truth and object detection results, respectively. TP and FP refer to true positive and false positive detection results. Number pair (u_{cls}, u_{reg}) gives classification and regression uncertainty scores. The uncertainty-regardless fusion method is still challenged by adverse conditions, whereas our fusion method remains robust.

2 Background

Object Detection based on Camera-LiDAR Fusion: According to the combination stage of LiDAR and camera data representations, current methods are categorized into data-, feature-, and proposal-level fusion. This paper focuses on proposal-level fusion for the following reasons. First, it is difficult to quantify uncertainty in data- and feature-level fusion because existing studies estimate uncertainty based on prediction proposals. Second, from the literature [23, 12, 24], proposal-level camera-LiDAR fusion achieves competitive and even superior performance compared with data- and feature-level fusion methods. Lastly, proposal-level fusion can easily incorporate alternative neural networks into the fusion pipeline and thus allowing easy adaptation to new object detector designs.

Predictive Uncertainty Estimation: Given plausible inputs, predictive uncertainty measures the variability of model predictions [15]. Predictive uncertainty includes *data uncertainty* due to observation noises caused by sensor measurements or environment and *model uncertainty* accounting for the uncertainty of the model that can be reduced by observing enough data. Traditionally, data and model uncertainties are modeled under the Bayesian deep learning framework. Specifically, given a data sample $\vec{x}$, the predictive uncertainty is $p(\hat{y}|\vec{x}, \mathcal{D}) = \int p(\hat{y}|\vec{x}, \vec{w})p(\vec{w}|\mathcal{D})d\vec{w}$, where $\mathcal{D}$ denotes the training dataset, $\vec{w}$ represents the model weights, $p(\vec{w}|\mathcal{D})$ and $p(\hat{y}|\vec{x}, \vec{w})$ characterize the model and data uncertainties.

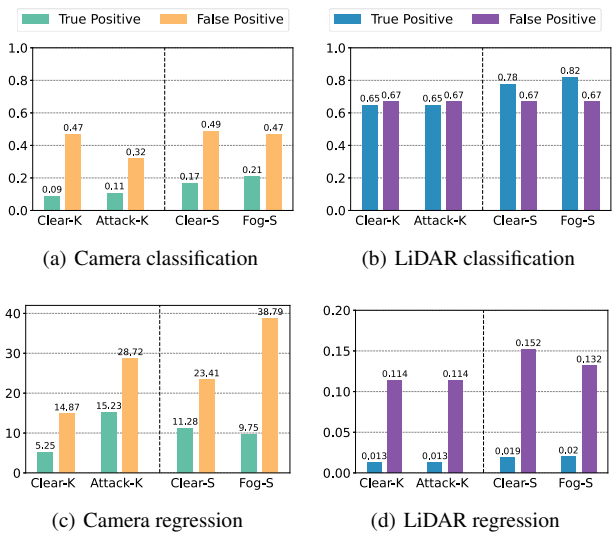
Data uncertainty is often modeled by Direct Modeling [8], which assumes that the model prediction follows a probability distribution and directly predicts the parameters of such distribution using the network output layers. Model uncertainty is usually approximated using techniques such as Monte Carlo (MC) Dropout [9] and deep ensembles [17], because it is intractable to calculate the weight posterior distribution over the dataset due to vast dimensionality. MC Dropout interprets dropout as a Bayesian approximation of deep Gaussian process. The model uncertainty is given by performing N forward passes on the same input with dropout enabled: $p(\hat{y}|\vec{x}, \mathcal{D}) \approx \frac{1}{N}\sum_{n=1}^{N} p(\hat{y}|\vec{x}, \vec{w})$. Deep ensemble estimates the predictive probability using an ensemble of models which have the same architecture and are trained with random initializations and data shuffled. Since deep ensemble incurs excessive memory footprint, in this paper, we adopt MC Dropout to estimate model uncertainty.

In AD, the object detector usually produces a bounding box for each detected object to describe the object location and the semantic category (e.g., car, pedestrian) with a probability score. While, the predicted bounding box regression variables are deterministic, and the probability score may not effectively characterize the classification uncertainties. Probabilistic object detectors aim to detect objects accurately and apply reliable uncertainty estimation in both classification and bounding box regression tasks, which additionally generate the *classification uncertainty* which is quantified by the probability that the object belongs to the target class and the *regression uncertainty* which is evaluated by the variance of the probability distribution over the predicted bounding box. The latter indicates the amount of uncertainty in the position of the box corners. Each of the classification and regression uncertainties includes data and model uncertainties.

Multi-modal Fusion based on Uncertainty: Multi-modal fusion considering the inherent uncertainty of individual modalities has been explored in previous literature. [27] estimates predictive uncertainty via variational inference across audio and visual modalities for the activity recognition task. Their approach seeks an optimal uncertainty threshold and it fuses predictive distributions that fall below

this threshold using average pooling. However, it restricts its consideration to information from non-degraded modalities with low uncertainty. Similarly, [29] merges multiple uncertainty metrics by applying a Min operation on their deviation ratios with respect to the training set. These ratios are subsequently utilized as the "temperature" for calibrating the prediction logit, and the logits are fused via the noisy-or operation. Nonetheless, this approach is primarily designed for the classification task, despite the fact that both classification and regression results serve as crucial indicators in a 3D object detection task. [4] fuses detection scores from multiple modalities employing a probabilistic approach based on Bayes' rule, coupled with the weighted average of boxes based on their data uncertainties. However, the method by [4] necessitates conditional independence across modalities and struggles to adapt to the 3D object detection task given that the output representations from two modalities differ. In contrast, our method adaptively fuse LiDAR and camera detections, harnessing both classification and regression uncertainties across all levels. It effectively addresses the representation disparity between LiDAR and camera detectors and possesses the flexibility to accommodate detectors with varying cognitive abilities.

3 Motivating Examples



(a) Camera classification (b) LiDAR classification

(c) Camera regression (d) LiDAR regression

Figure 2. Average uncertainty scores in four different driving contexts: clear and synthetic adversarial attack scenes from KITTI dataset; clear and dense fog scenes from STF dataset. The vertical dashed line separates the two different datasets.

For object detection in real driving scenarios, the classification and regression uncertainties are affected by variations in the environment or sensor data and the detector's cognitive ability determined by $\vec{w}$. For example, adverse weather and adversarial attacks can degrade the detection performance and increase the uncertainties. This section provides motivating examples to understand how the predictive uncertainties of LiDAR and camera vary in normal and adverse scenarios. We also preview the advantages of our proposed uncertainty-encoded LiDAR-camera fusion.

Fig. 1 shows the detection results of the input frames under four representative driving contexts: (a) the clear weather scene from the KITTI dataset [10], (b) the scene under the adversarial perturbation attack [20] against camera, (c) clear scenes, and (d) dense fog weather scenes from the STF dataset [1]. We compare the detection performance of the camera-based detector RetinaNet [19], LiDAR-based detector SECOND [33], uncertainty-regardless LiDAR-camera fusion detector CLOCs [23], and our uncertainty-encoded LiDAR-camera fusion detector. The detection results in the last three columns are shown in bird's-eye view.

In this section, we use Eqs. (1) and (3) presented later to estimate the scalar classification and regression uncertainty scores. In a more extensive set of experiments, we compute the average classification and regression uncertainty scores of true positive and false positive detections for LiDAR and camera over datasets of the aforementioned driving scenes. False positive detections can lead to incorrect evasive actions that may cause accidents. The results are presented in Fig. 2.

Fig. 1 and Fig. 2 give four observations. First, classification and regression uncertainty scores, especially for false positives, generally increase when sensors are affected by adverse scenarios. Second, in the same driving scenes, false positives generate much higher classification and regression uncertainty scores than true positives. However, LiDAR tends to produce higher classification uncertainty for true positives in adverse weather condition of Fig. 2(b), as a result of the lower quality and challenging nature of the dataset used. This highlights the need for utilizing both classification and regression uncertainty to address deficiencies in the dataset. Third, camera and LiDAR have different sensitivities and cognitive abilities to the environment changes. Camera's uncertainty scores show greater volatility than LiDAR's. Lastly, camera's and LiDAR's regression uncertainty values are in different orders.

From above, predictive uncertainty is indicative of sensing performance, while camera's and LiDAR's uncertainties exhibit distinct sensitivities under different adverse scenarios. This motivates us to design a new fusion method with encoded uncertainties as part of the input. As previewed by the last column of Fig. 1, our method effectively exploits uncertainties for robust object detection under adverse conditions.

4 Problem Formulation

This paper considers fusion-based object detection using camera and LiDAR. The 3D point cloud data from LiDAR, denoted by $\vec{x}^L \in \mathbb{R}^3$, and the 2D RGB image data from camera, denoted by $\vec{x}^I \in \mathbb{R}^2$, are processed by the LiDAR detection branch H_L and camera detection branch H_I, respectively. For each frame, H_L produces detection $\vec{p}^L = \{\vec{p}^L_1, \ldots, \vec{p}^L_{M_L}\}$, where each $\vec{p}^L_{m_L}$ represents a proposal consisting of the 3D bounding box coordinates and the confidence score. Similarly, H_I generates detection $\vec{p}^I = \{\vec{p}^I_1, \ldots, \vec{p}^I_{M_I}\}$. The fusion combining H_L and H_I produces detection $\vec{p} = \{\vec{p}_1, \ldots, \vec{p}_K\}$, where $K = M_L \times M_I$ and each element $\vec{p}_k = (\vec{p}^L_k, \vec{p}^I_k)$ is a proposal pair consisting of a LiDAR proposal and a camera proposal. We aim to explicitly use LiDAR's and camera's detection uncertainties in the above fusion process to derive the final detection that is robust under various adverse conditions covered by training data.

To this end, we first employ uncertainty estimation for LiDAR and camera proposals to derive LiDAR detection uncertainty $\vec{u}^L = \{\vec{u}^L_1, \ldots, \vec{u}^L_{M_L}\}$ and camera detection uncertainty $\vec{u}^I = \{\vec{u}^I_1, \ldots, \vec{u}^I_{M_I}\}$. Then, we aim to derive the weights that determine the importance of the two sensing modalities for each proposal pair $(\vec{p}^L_k, \vec{p}^I_k) \in \vec{p}$ based on uncertainties $\vec{u}^L$ and $\vec{u}^I$. To preserve the consistency of the input for subsequent fusion models, we implement the weights by refining the confidence score of detection result. Specifically, we aim to find the function represented by f_u that maps LiDAR and camera proposals and their uncertainty to uncertainty-

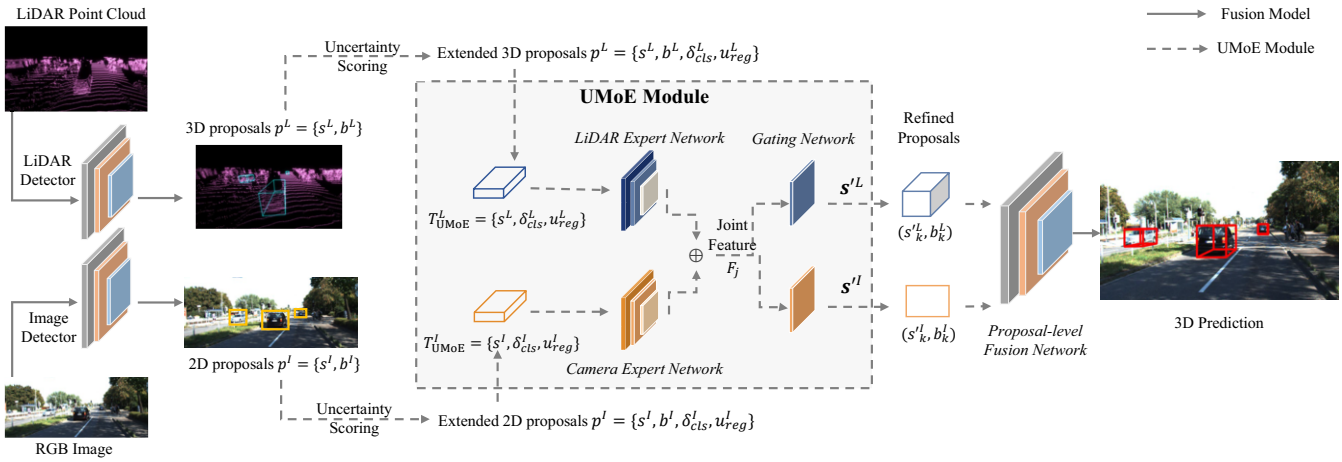


Figure 3. The pipeline of the UMoE module integrated proposal-level LiDAR-camera fusion network. Dotted lines represent data flow of the UMoE module, while solid lines are for general proposal-level fusion.

encoded confidence score $\vec{s'}$, i.e., $\vec{s'} = f_u(\vec{p}^L, \vec{u}^L, \vec{p}^I, \vec{u}^I)$, where $\vec{s'} = \{\vec{s'^L}, \vec{s'^I}\}$. Subsequently, proposals that have their confidence scores replaced by $\vec{s'}$ can be fused to generate the final detection: $\vec{p}^* = f_s(\vec{p}^L, \vec{p}^I)$, where $f_s(\cdot)$ is the fusion operation or fusion network.

There are three main challenges in designing f_u. First, f_u requires informative and practical uncertainty representations $\vec{u}^L$ and $\vec{u}^I$ as input. The representation should be distinguishable, ensuring that it assigns different values to true positives and false positives across all scenarios. Besides, in the context of AD, the uncertainty representation should be calculated without any groundtruth labels and computationally efficient. Second, the two modalities' uncertainties respond to adverse conditions differently. For instance, in the fog scenarios of Figs. 2(c) and 2(d), the average regression uncertainty score for false positives from camera increases significantly, while the value for LiDAR decreases slightly. Third, regression uncertainties computed based on 3D and 2D bounding boxes from LiDAR and camera lie in different ranges. The common 2D bounding box representation encodes top-left and bottom-right corners in camera coordinate, while 3D bounding box includes position, dimension, and rotation angle in LiDAR coordinate. The range difference of these encoded elements results in higher regression uncertainty for a 2D bounding box than that of a 3D bounding box. Our results in Fig. 2(c) and 2(d) exhibit this issue. The above discrepancies render straightforward ways of incorporating uncertainties, e.g., admitting raw uncertainties as fusion input, futile.

5 Uncertainty-encoded Mixture-of-Experts (UMoE) Fusion Module

To address the above challenges, we propose a fusion module called Uncertainty-encoded Mixture-of-Experts (UMoE) that bridges the sensor-specific detectors and the proposal-level fusion network. Fig. 3 illustrates the proposal-level LiDAR-camera fusion framework with our UMoE module integrated. First, LiDAR- and camera-based detectors take sensor data to generate detection proposals. With uncertainty scoring, we extend each proposal with uncertainty scores to build the UMoE input. Then, the expert network for each sensing modality extracts sensor-specific features from the UMoE input. The gating network takes the combined features generated by the preceding expert networks and generates the uncertainty-encoded confidence scores. Finally, detection proposals with updated confidence scores can be applied in the proposal-level fusion networks.

5.1 Uncertainty Scoring

Now we describe our uncertainty scoring approach. Denote object proposals produced by sensor-specific detectors as $\vec{p} = \{s, \vec{b}\}$, where s and $\vec{b}$ denote confidence score and bounding box. By using the MC Dropout uncertainty estimation approach, each proposal $\vec{p}_k$ is assigned with uncertainty $\vec{u}_k = \{\vec{u}_{k,cls}, \vec{u}_{k,reg}\}$ consisting of the classification uncertainty $\vec{u}_{k,cls} \in \mathbb{R}^C$ and regression uncertainty $\vec{u}_{k,reg} \in \mathbb{R}^B$, where C and B are the numbers of classes and elements in bounding box representation. To encode $\vec{u}_k$ into UMoE input tensor, we transform the vectors $\vec{u}_{k,cls}$ and $\vec{u}_{k,reg}$ into scalar uncertainty scores $u_{k,cls} \in \mathbb{R}$ and $u_{k,reg} \in \mathbb{R}$ as follows.

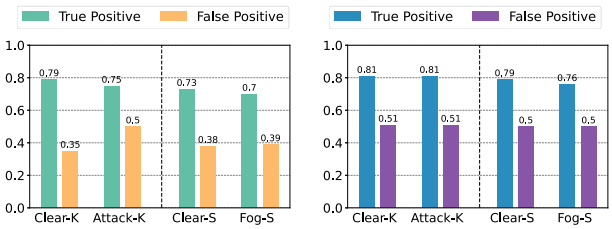
First, we use entropy to score classification uncertainty:

$$u_{k,cls} = -\sum_{c=1}^{C} s_c \log s_c, \tag{1}$$

where $s_c = \frac{1}{N}\sum_{n=1}^{N} p(\hat{y} = c|\vec{x}_k, \vec{w}_n)$ represents the average predicted classification probability of class c over the N forward passes of MC Dropout. Eq. 1 yields high classification uncertainty scores $u_{k,cls}$ for proposals with intermediate average predicted classification probability s_c, while demonstrating reduced values for proposals with s_c at either extreme. However, the task complexity is modality-dependent, with LiDAR-based 3D detectors generally exhibit lower confidence levels compared to their camera-based 2D counterparts. Consequently, as illustrated in Fig. 2(b) for LiDAR classification uncertainty scores, false positives with low confidence scores have smaller average scores than true positives. To ensure the informativeness of the classification uncertainty score, we enhance it with a classification deviation ratio, a quantitative metric designed to evaluate the extent to which a proposal's confidence score and classification uncertainty score deviate from the true positives' distribution. Specifically:

$$\delta_{k,cls} = \frac{\mu_u}{\mu_u + \max(0, (u_{k,cls} - \mu_u - \sigma_u))} \cdot \frac{\mu_s}{\mu_s + \max(0, -(s_c - \mu_s - \sigma_s))} \tag{2}$$

where $\mu_u, \sigma_u, \mu_s, \sigma_s$ are the mean and standard deviation of classification uncertainty scores and the average predicted classification probability for true positives in the validation set. The computed ratio, $\delta_{k,cls}$, serves to assign larger value to proposals whose classification uncertainty score $u_{k,cls}$ and average predicted classification probability s_k fall within the distribution of true positives, while assigning smaller values to those proposals that are out-of-distribution. As illustrated in Fig. 4, the deviation ratio is more informative than the classification uncertainty score, as it effectively captures the differences between false positives and true positives, as well as accounting for adverse conditions.



(a) Camera deviation ratio (b) LiDAR deviation ratio

Figure 4. Average cls deviation ratio in the same contexts as Fig. 2.

After that, we use the total variance of $\vec{u}_{k,reg}$ to score regression uncertainty:

$$u_{k,reg} = \mathrm{tr}\left(\frac{1}{N}\sum_{n=1}^{N}\vec{b}_{k,n}\vec{b}_{k,n}^{\intercal} - \vec{\bar{b}}_k\vec{\bar{b}}_k^{\intercal}\right), \tag{3}$$

where $\vec{\bar{b}}_k = \frac{1}{N}\sum_{n=1}^{N}\vec{b}_{k,n}$ is the mean bounding box coordinates over the N forward passes of MC Dropout. On top of this, we adopt the Direct Modeling method for data uncertainty of bounding boxes. We assume that each regression output follows an independent Gaussian distribution and estimate the variance of these outputs. We then generate Monte Carlo samples from this distribution and add the total variance of these samples to $u_{k,reg}$. The total variance ranges in $[0, \infty]$, where a larger value indicates higher regression uncertainty. However, the sizes of the 2D bounding boxes in the image plane have significant influence on the values of the total variance. For instance, a closer object with a larger bounding box leads to a large $u_{k,reg}$. To achieve fair comparisons among objects at different distances, we divide $u_{k,reg}$ by the diagonal length of the averaged bounding box $\vec{\bar{b}}_k$. We also apply standardization to $u_{k,reg}$ with mean and standard deviation of regression uncertainty calculated from the clear validation set.

With the uncertainty scores, we extend each original proposal $\vec{p}_k = \{s_k, \vec{b}_k\}$ for both LiDAR and camera to $\vec{p}_k = \{s_k, \vec{b}_k, \delta_{k,cls}, u_{k,reg}\}$. For a given scene, all of the K LiDAR-camera proposal pairs are used to build the tensors $\vec{T}^I_{\mathrm{UMoE}} = \{\vec{s}^I, \vec{\delta}^I_{cls}, \vec{u}^I_{reg}\} \in \mathbb{R}^{1\times K\times 3}$, $\vec{T}^L_{\mathrm{UMoE}} = \{\vec{s}^L, \vec{\delta}^L_{cls}, \vec{u}^L_{reg}\} \in \mathbb{R}^{1\times K\times 3}$, which will be used as input to the UMoE module.

5.2 Multi-Modal Uncertainty Fusion via UMoE

Our UMoE is based on the Mixture of Experts (MoE) architecture [14], which is designed to handle multiple different tasks in complex scenarios. Existing works [16, 21] demonstrate MoE's effectiveness in multi-modal perception including LiDAR-camera fusion. The key advantage of MoE is that it contains distinct expert networks to extract features from each of the sensing modalities and then uses a gating network to combine and learn from the different extracted features to give final output. To find the mapping f_u, our UMoE module substitutes the traditional inputs of MoE with $\vec{T}^I_{\mathrm{UMoE}}$ and $\vec{T}^L_{\mathrm{UMoE}}$ to produce uncertainty-encoded confidence scores $\vec{s'}$. The detailed designs of our UMoE components are as follows.

5.2.1 Expert networks

As shown in the motivating example section, different sensing modalities have distinct sensitivities and value ranges for uncertainty scoring. Thus, we exploit different expert network for each sensing modality that maps input tensors to sensor-specific features for further fusion. Specifically, the expert network for LiDAR branch E_L consists of a set of Residual blocks [11]. The Residual Block operation is denoted by $\mathrm{ResBlock}(c_{in}, c_{out}, k)$, where c_{in}, c_{out} are the input and output channel size and k is the kernel size of 2D convolution layers inside. In E_L, we employ three Residual blocks $\mathrm{ResBlock}(3, 9, (1, 1))$, $\mathrm{ResBlock}(9, 18, (1, 1))$, and $\mathrm{ResBlock}(18, 18, (1, 1))$ sequentially. Similarly, the expert network for camera branch E_I follows the same structure. The process can be described as: $\vec{F}^L = E_L(\vec{T}^L_{\mathrm{UMoE}})$, $\vec{F}^I = E_I(\vec{T}^I_{\mathrm{UMoE}})$, where $\vec{F}^L, \vec{F}^I \in R^{1\times K\times 18}$ are the feature vectors that encode confidence score and uncertainties for each proposal of the corresponding sensing modality.

5.2.2 Gating network

The gating network concatenates features $\vec{F}^L$, $\vec{F}^I$ across all sensor modalities into a joint feature $\vec{F}_J$, which yields a tensor with size $1 \times K \times 36$. Next, two output branches $G^L(\cdot)$ and $G^I(\cdot)$ take the same joint feature F_J as input and predict uncertainty-encoded confidence scores $\vec{s'}^L$ and $\vec{s'}^I$, respectively. Each output branch consists of the a single Residual block $\mathrm{ResBlock}(36, 1, (1, 1))$. The pipeline of the gating network is defined as follows: $\vec{s'}^L = G^L(\vec{F}_J)$, $\vec{s'}^I = G^I(\vec{F}_J)$, where $\vec{F}_J = \vec{F}^L \oplus \vec{F}^I$. The outputs $\vec{s'}^L$ and $\vec{s'}^I$ are used to substitute the original confidence scores $\vec{s}^L$ and $\vec{s}^I$ of corresponding proposals based on their uncertainty scores. These refined proposals can then be input into various proposal-level fusion networks for further processing.

5.3 Training

We now present the training of our UMoE module. We first train the sensor-specific detectors and then fix them to train the UMoE module. For sensor-specific detectors, we add dropout layers to enable model uncertainty estimation using the MC-Dropout approach. Moreover, we follow the Direct Modeling approach to add the following loss function to the training of the sensor-specific detectors: $\mathcal{L}_{\mathrm{add}} = \frac{1}{2}\exp(-\log(\vec{\sigma}^2))||\vec{b_{gt}} - \vec{b}|| + \frac{1}{2}\log(\vec{\sigma}^2)$, where $\vec{\sigma}$ is the estimated data uncertainty, $\vec{b}$ is the predicted bounding box, and $\vec{b_{gt}}$ is the corresponding ground truth. With modified sensor-specific detectors, we can train the UMoE module solely or with a proposal-level fusion network in an end-to-end manner. In this way, the UMoE module learns to produce uncertainty-encoded confidence score via supervision from final detection $\vec{p}^*$, i.e., the 3D predictions generated by the fusion network.

6 Experiments

This section evaluates UMoE in comparison with uncertainty-regardless LiDAR-camera fusion methods on four datasets that cover the scenarios of clear/adverse weather conditions, adversarial attack, and camera blinding attack.

Table 1. AP_{3D} on KITTI (clear), KITTIAdv (attack) and KITTIBlind (attack) datasets w/o and w/ the UMoE module.

Method	UMoE	KITTI AP_{3D}			KITTIAdv AP_{3D}			KITTIBlind AP_{3D}		
		easy	mod.	hard	easy	mod.	hard	easy	mod.	hard
CLOCs_SecRetina		**91.61**	81.86	77.68	87.77	77.73	73.87	87.89	78.28	76.24
	✓	90.25	**81.87**	**79.02**	**89.36**	**77.82**	**75.32**	**90.43**	**81.80**	**79.08**
CLOCs_PointRetina		**90.42**	**81.80**	**78.62**	85.48	75.41	73.27	84.59	78.28	75.70
	✓	89.88	80.47	77.77	**88.65**	**77.82**	**75.55**	**89.99**	**80.31**	**77.81**

Table 2. AP_{3D} performance on STF clear, dense fog and snow test splits w/o and w/ the UMoE module.

Method	UMoE	Clear AP_{3D}			Dense Fog AP_{3D}			Snow AP_{3D}		
		easy	mod.	hard	easy	mod.	hard	easy	mod.	hard
CLOCs_SecRetina		**49.89**	47.97	**43.88**	31.55	31.92	28.10	43.52	41.14	36.43
	✓	49.84	**48.41**	43.40	**37.05**	**35.48**	**32.32**	**47.27**	**44.25**	**39.75**
CLOCs_PointRetina		**46.53**	43.73	39.73	28.67	27.80	27.34	**43.49**	40.16	**36.48**
	✓	46.37	**45.46**	**41.63**	**39.34**	**37.18**	**32.89**	43.04	**40.28**	36.20

6.1 Datasets

Clear Scenario - KITTI: For sunny daytime driving scenarios, we use KITTI 3D object detection dataset [10] and follow the standard data split [3] with a 3,712 frames *train* set. We randomly divide the standard val split into a *val* set with 1,884 frames and a *test* set with 1,885 frames. This allows us to calculate deviation ratio and enables the evaluation of subsequent attack datasets created based on *test* set.

Adversarial Attack Scenario - KITTIAdv: We synthesize the adversarial attack scenario by perturbing camera images using PGD [20] method on the divided KITTI *test* set (See details in Appendix A.1[1]). The attack strength of PGD attack is $4/255$.

Camera Blind Attack Scenario - KITTIBlind: We follow methods in [35] to generate facula with a radius of 112 pixel and overlay it on the divided KITTI *test* set to form our self-synthetic KITTIBlind dataset that mimics strong light exposure affecting the camera modality (details are described in Appendix A.1).

Adverse Weather Scenario - STF: To simulate adverse weather, we adopt the STF [2] dataset including clear, dense fog and snow scenarios. The STF clear weather training set, validation set and test set has 3686, 921 and 1536 scenes. Its dense fog test set has 88 scenes; snow test set has 1161 scenes. For models trained on clear weather training set, we select the snapshot with the best performance on clear validation set and evaluate it on aforementioned test splits.

Table 3. Comparison AP_{3D} results of the proposed method and the state-of-the-art fusion baselines in KITTI and KITTIBlind (attack) datasets. We highlight the best performance in bold and the second best in underline.

Dataset	Method	AP_{3D}		
		easy	mod.	hard
KITTI	PointPainting	89.23	79.31	76.86
	EPNet	**92.16**	**82.69**	**80.10**
	CLOCs	<u>91.61</u>	81.86	77.68
	CLOCs + UMoE	90.25	<u>81.87</u>	<u>79.02</u>
KITTIBlind	PointPainting	87.26	77.20	74.44
	EPNet	87.03	75.38	73.78
	CLOCs	<u>87.89</u>	<u>78.28</u>	<u>76.24</u>
	CLOCs + UMoE	**90.43**	**81.80**	**79.08**

[1] The appendix can be found online at https://arxiv.org/abs/2307.16121.

6.2 Implementation

This section describes the implementation of our approach and the employed baselines.

Uncertainty-regardless fusion baseline: We select the CLOCs [23], PointPainting [30] and EPNet [13] as representative proposal-level, data-level and feature-level fusion baselines. For the CLOCs, we combine SECOND [33] and RetinaNet [19], named CLOCs_SecRetina, and adopt PointPillar [18] and RetinaNet, named CLOCs_PointRetina as the 3D and 2D detectors. We use OpenPCDet [28] and Detectron2 [32] as our 3D and 2D codebases and apply the default settings. For the CLOCs fusion network, we follow [24] that use Residual blocks instead of standard 1×1 convolution layers and optimized with Adam optimizer for 20 epochs. We employed the OneCycleLR learning rate scheduler with an initial learning rate of 6×10^{-5}, a maximum learning rate of 6×10^{-4}, and a weight decay of 0.01. A specific description of the CLOCs fusion model structure is detailed further in Appendix A.2. For PointPainting and EPNet, we fork the original implementations without any modification.

Uncertainty-encoded fusion: We integrate the UMoE module into CLOCs_SecRetina and CLOCs_PointRetina as our uncertainty-encoded fusion models. To enable uncertainty estimation on sensor-specific detectors, we follow two steps to retrain 3D and 2D detectors. First, we add dropout after each DeConv2D layer for the 3D detectors and after each Conv2D layer for the detection head of the RetinaNet. The dropout rate is set at 0.1 for all the detectors. Next, the additional loss item previously mentioned is added during training to estimate data uncertainty (refer to Appendix A.2 for explicit sensor-specific detectors' training settings). During the inference of sensor-specific detectors, we perform 10 stochastic samplings with dropout enabled, as suggested in [15]. With the retrained sensor-specific detectors fixed, we apply the same settings with the CLOCs fusion network and train our UMoE module with the fusion network in an end-to-end manner.

6.3 Overall Performance

We report our evaluation results on the most dominant class, cars, in four datasets. The Average Precision of 40 recall position with an IoU threshold of 0.7 in 3D space (AP_{3D}) is used as the evaluation metrics. Due to the space limitation, we present visualization figures in Appendix A.4.

Table 4. Ablation study about the effects of classification deviation ratio (DR.), regression uncertainty score (Reg.) and the MoE architecture. The best results are in bold and the second best are underlined.

DR.	Reg.	MoE	KITTIAdv			KITTIBlind			STF Snow			STF Dense Fog		
			easy	mod.	hard	easy	mod.	hard	easy	mod.	hard	easy	mod.	hard
	✓	✓	88.91	77.21	74.87	<u>90.07</u>	<u>81.59</u>	<u>79.02</u>	<u>45.91</u>	<u>42.90</u>	<u>38.99</u>	<u>36.58</u>	35.00	<u>32.12</u>
✓		✓	<u>89.13</u>	<u>77.76</u>	<u>75.31</u>	89.96	81.56	78.91	45.14	42.51	38.56	34.50	34.65	30.77
✓	✓		85.81	75.57	73.82	87.53	80.27	78.18	45.59	42.76	38.87	35.55	<u>35.37</u>	31.18
✓	✓	✓	**89.36**	**77.82**	**75.32**	**90.43**	**81.80**	**79.08**	**47.27**	**44.25**	**39.75**	**37.05**	**35.48**	**32.32**

KITTI: Based on evaluation results on KITTI *test* set, our UMoE module maintains a satisfactory overall performance under clear scenarios. The UMoE-integrated fusion model performs comparably to uncertainty-regardless fusion in Table 1. Similar observations can be found in Table 3 when comparing with state-of-the-art fusion baselines.

KITTIAdv: With numerous false positives generated from the camera-based detector due to the adversarial attack, AP_{3D} of uncertainty-regardless fusion baseline drops rapidly. However, the fusion models integrated with UMoE outperform the baseline, i.e., improve AP_{3D} from 87.77% to 89.36% on easy objects. This result may be attributed to the fact that UMoE down-weights false positive proposals with large uncertainty.

KITTIBlind: As seen in the KITTIBlind dataset results presented in Table 1, our UMoE module outperforms the uncertainty-regardless fusion baseline significantly and maintains a similar level of performance as in clear scenarios. Additionally, the AP_{3D} of all state-of-the-art fusion baselines decrease with affected camera under strong light exposure in Table 3, while our module remains robust.

STF: We show the evaluation results for the STF dataset in Table 2. LiDAR and camera degrade simultaneously in extreme weather conditions. The AP_{3D} for both fusion models decreases in these scenes due to noisy LiDAR point clouds and camera images. Our proposed UMoE module significantly improves the AP_{3D} under adverse weather scenarios, with a maximum increase of 10.67% under dense fog weather and 3.75% in snow scenes. Additionally, the UMoE-integrated fusion models achieve comparable or even better results in clear scenarios. These results demonstrate that UMoE can effectively improve robustness in extreme weather conditions.

Statistical significant test: We calculate p-values on moderate AP_{3D} from 10 runs of CLOCs_SecRetina and its baseline, which are 0.01, 2.9×10^{-14}, 1.4×10^{-7}, 1.6×10^{-8} in KITTIAdv, KITTIBlind, snow and dense fog scenarios. Each p-value is less than 0.05, confirming the significance of our module's improvement.

6.4 Ablation Study

To analyze the effects of uncertainty scoring and the MoE architecture, we conduct ablation studies by removing each component on the CLOCs_SecRetina model. We report results in Table 4, including AP_{3D} on KITTIAdv dataset, KITTIBlind dataset, and the snow and dense fog test sets from the STF dataset.

Encoded uncertainty: To study the effectiveness of the encoded uncertainties, we remove classification deviation ratio or regression uncertainty scores from the input tensor $\vec{T}^{I}_{\text{UMoE}}$ and $\vec{T}^{L}_{\text{UMoE}}$. As shown in Table 4, utilizing only the classification deviation ratio (row 2) provides relatively limited benefits, though it is particularly advantageous in adversarial attack scenes. Conversely, relying solely on the regression uncertainty score (row 1) generally results in more considerable benefits, especially in dense fog scenarios, likely attributable to its effectiveness in identifying false positives. Moreover, incorporating all components (row 4) culminates in the highest performance across all scenarios, suggesting that both uncertainties serve as crucial cues in the 3D object detection task.

MoE: To investigate the effectiveness of MoE, we remove this architecture and feed uncertainty scores directly to the fusion layer. The results in row 3 demonstrate that even with complete uncertainties, the model without MoE performs poorly in some adversarial attack and snow scenarios. This confirms the necessity of using MoE in handling the uncertainty differences across modalities and ranges.

6.5 MC-dropout runtime analysis

This section briefly analyzes the runtime of the MC-dropout technique applied in our method. As described in Section 6.2, we perform 10 MC-dropout runs only on the detection head of sensor-specific detectors during the inference. Under these settings, the running speed is around 6 fps and 40 fps for uncertainty-encoded and uncertianty-regardless RetinaNet, respectively. The speeds are 11 fps and 24 fps for SECOND detector and 15 fps and 40 fps for PointPillar. It is worth noting that our extensive experiments show that the detection performance growth stabilizes after 5 runs. With the development of edge devices, computational redundancy can be exploited when using MC-dropout. Therefore, it will not drastically increase the cost.

6.6 Limitations

Our approach encounters two primary limitations. Firstly, sensor-specific detectors may suffer slight performance degradation due to uncertainty estimation techniques such as MC-Dropout, particularly in clear scenarios, which can impact fusion performance. However, our approach can adapt advanced uncertainty estimation method to minimize such reductions. Secondly, the UMoE module can only mitigate, not eliminate, the effect of adverse scenarios. In situations where all sensors fail, our method's enhancement may be limited.

7 Conclusion

Autonomous driving is moving rapidly toward a higher level of automation in more complex environments, demanding the ability of coping with all kinds of uncertainties. This paper systematically studied how to incorporate the predictive uncertainties of individual sensors in multi-modal fusion, a fundamental task of autonomous driving perception. We score uncertainties and propose a fusion module that exploits the Mixture-of-Expert architecture to encode multi-modal uncertainties in any proposal-level fusion pipelines. Experimental results show that our module significantly improves the fusion performance in adverse scenarios. In addition to LiDAR-camera fusion, the scope of our methods can be broadened to encompass various scenarios, like incorporating additional sensors such as radar, or enhancing the LiDAR-only single-modality detection which is common in industrial-level autonomous driving systems. Far beyond the object detection metrics, evaluating the robustness of multi-modal fusion in various downstream tasks is interesting for future work.

Acknowledgements

The paper is partially supported by Hong Kong Research Grant Council under GRF project 11210622.
This research is supported by the Ministry of Education, Singapore, under its Academic Research Fund Tier 1 (RT14/22).

References

[1] Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide, 'Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11682–11692, (2020).

[2] Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide, 'Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (June 2020).

[3] 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*, pp. 1907–1915, (2017).

[4] Yi-Ting Chen, Jinghao Shi, Zelin Ye, Christoph Mertz, Deva Ramanan, and Shu Kong, 'Multimodal object detection via probabilistic ensembling', in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IX*, pp. 139–158. Springer, (2022).

[5] Di Feng, Yifan Cao, Lars Rosenbaum, Fabian Timm, and Klaus Dietmayer, 'Leveraging uncertainties for deep multi-modal object detection in autonomous driving', in *2020 IEEE Intelligent Vehicles Symposium (IV)*, pp. 877–884. IEEE, (2020).

[6] Di Feng, Christian Haase-Schütz, Lars Rosenbaum, Heinz Hertlein, Claudius Glaeser, Fabian Timm, Werner Wiesbeck, and Klaus Dietmayer, 'Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges', *IEEE Transactions on Intelligent Transportation Systems*, **22**(3), 1341–1360, (2020).

[7] Di Feng, Lars Rosenbaum, and Klaus Dietmayer, 'Towards safe autonomous driving: Capture uncertainty in the deep neural network for lidar 3d vehicle detection', in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 3266–3273. IEEE, (2018).

[8] Yarin Gal, *Uncertainty in deep learning*, Ph.D. dissertation, University of Cambridge, 2016.

[9] Yarin Gal and Zoubin Ghahramani, 'Dropout as a bayesian approximation: Representing model uncertainty in deep learning', in *international conference on machine learning*, pp. 1050–1059. PMLR, (2016).

[10] Andreas Geiger, Philip Lenz, and Raquel Urtasun, 'Are we ready for autonomous driving? the kitti vision benchmark suite', in *2012 IEEE conference on computer vision and pattern recognition*, pp. 3354–3361. IEEE, (2012).

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, 'Deep residual learning for image recognition', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, (2016).

[12] Keli Huang, Botian Shi, Xiang Li, Xin Li, Siyuan Huang, and Yikang Li, 'Multi-modal sensor fusion for auto driving perception: A survey', *arXiv preprint arXiv:2202.02703*, (2022).

[13] Tengteng Huang, Zhe Liu, Xiwu Chen, and Xiang Bai, 'Epnet: Enhancing point features with image semantics for 3d object detection', in *European Conference on Computer Vision*, pp. 35–52. Springer, (2020).

[14] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton, 'Adaptive mixtures of local experts', *Neural computation*, **3**(1), 79–87, (1991).

[15] Alex Kendall and Yarin Gal, 'What uncertainties do we need in bayesian deep learning for computer vision?', *Advances in neural information processing systems*, **30**, (2017).

[16] Jaekyum Kim, Junho Koh, Yecheol Kim, Jaehyung Choi, Youngbae Hwang, and Jun Won Choi, 'Robust deep multi-modal learning based on gated information fusion network', in *Asian Conference on Computer Vision*, pp. 90–106. Springer, (2018).

[17] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell, 'Simple and scalable predictive uncertainty estimation using deep ensembles', *Advances in neural information processing systems*, **30**, (2017).

[18] 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*, pp. 12697–12705, (2019).

[19] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár, 'Focal loss for dense object detection', in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, (2017).

[20] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu, 'Towards deep learning models resistant to adversarial attacks', in *International Conference on Learning Representations*, (2018).

[21] Oier Mees, Andreas Eitel, and Wolfram Burgard, 'Choosing smartly: Adaptive multimodal fusion for object detection in changing environments', in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 151–156. IEEE, (2016).

[22] Dimity Miller, Feras Dayoub, Michael Milford, and Niko Sünderhauf, 'Evaluating merging strategies for sampling-based uncertainty techniques in object detection', in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 2348–2354. IEEE, (2019).

[23] Su Pang, Daniel Morris, and Hayder Radha, 'Clocs: Camera-lidar object candidates fusion for 3d object detection', in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 10386–10393. IEEE, (2020).

[24] Su Pang, Daniel Morris, and Hayder Radha, 'Fast-clocs: Fast camera-lidar object candidates fusion for 3d object detection', in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 187–196, (2022).

[25] Pony.ai. https://pony.ai/, 2022.

[26] Vishwanath A Sindagi, Yin Zhou, and Oncel Tuzel, 'Mvx-net: Multimodal voxelnet for 3d object detection', in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 7276–7282. IEEE, (2019).

[27] Mahesh Subedar, Ranganath Krishnan, Paulo Lopez Meyer, Omesh Tickoo, and Jonathan Huang, 'Uncertainty-aware audiovisual activity recognition using deep bayesian variational inference', in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6301–6310, (2019).

[28] OpenPCDet Development Team. Openpcdet: An open-source toolbox for 3d object detection from point clouds. https://github.com/open-mmlab/OpenPCDet, 2020.

[29] Junjiao Tian, Wesley Cheung, Nathaniel Glaser, Yen-Cheng Liu, and Zsolt Kira, 'Uno: Uncertainty-aware noisy-or multimodal fusion for unanticipated input degradation', in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5716–5723. IEEE, (2020).

[30] 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*, pp. 4604–4612, (2020).

[31] Waymo. https://waymo.com/, 2022.

[32] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

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

[34] Jin Hyeok Yoo, Yecheol Kim, Jisong Kim, and Jun Won Choi, '3d-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection', in *European Conference on Computer Vision*, pp. 720–736. Springer, (2020).

[35] Jindi Zhang, Yifan Zhang, Kejie Lu, Jianping Wang, Kui Wu, Xiaohua Jia, and Bin Liu, 'Detecting and identifying optical signal attacks on autonomous driving systems', *IEEE Internet of Things Journal*, **8**(2), 1140–1153, (2020).